Cooperative perception : Application in the context of outdoor intelligent vehicle systems

Hao Li

▶ **To cite this version:**

Hao Li. Cooperative perception : Application in the context of outdoor intelligent vehicle systems. Other. Ecole Nationale Supérieure des Mines de Paris, 2012. English. NNT : 2012ENMP0034 . pastel-00766986

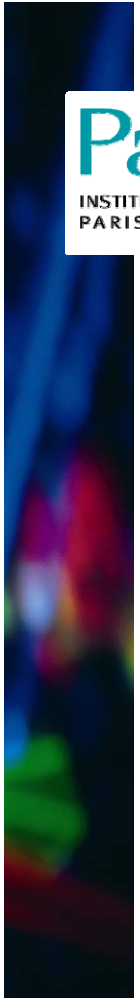
HAL Id: pastel-00766986

https://pastel.archives-ouvertes.fr/pastel-00766986

Submitted on 19 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

École doctorale n° 432 : Sciences des Métiers de l'Ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure des Mines de Paris

Spécialité "Informatique temps réel, Robotique, Automatique"

présentée et soutenue publiquement par

Hao LI

le 21 septembre 2012

Cooperative Perception:

Application in the Context of Outdoor Intelligent Vehicle Systems

(Perception Coopérative : Application au Contexte des Systèmes de Véhicules Intelligents à l'Extérieur)

Directeur de thèse : **Fawzi NASHASHIBI**

Jury

M. Roland CHAPUIS,	Professeur	Polytech Clermont-Ferrand	Rapporteur
M. Philippe BONNIFAIT	Professeur	UTC Compiegne	Rapporteur
M. Christian LAUGIER	DR1/Thèse d'Etat	INRIA Rhône-Alpes	Président
M. Bruno STEUX	Docteur	Mines Paristech	Examinateur
M. Michel PARENT	Docteur	INRIA Paris-Rocquencourt	Examinateur
M. Fawzi NASHASHIBI,	Directeur de Recherche	INRIA Paris-Rocquencourt	Examinateur

T H È S E

MINES ParisTech
Centre de Robotique
60 Bd St. Michel 75006 Paris

ABSTRACT

The research theme of this dissertation is the multiple-vehicles cooperative perception (or *cooperative perception*) applied in the context of intelligent vehicle systems. The general methodology of the presented works in this dissertation is to realize multiple-intelligent vehicles cooperative perception, which aims at providing better vehicle perception result compared with single vehicle perception (or *non-cooperative perception*). Instead of focusing our research works on the absolute performance of cooperative perception, we focus on the general mechanisms which enable the realization of cooperative localization and cooperative mapping (and moving objects detection), considering that localization and mapping are two underlying tasks for an intelligent vehicle system. We also exploit the possibility to realize certain augmented reality effect with the help of basic cooperative perception functionalities; we name this kind of practice as *cooperative augmented reality*. Naturally, the contributions of the presented works consist in three aspects: cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality.

Description

We have used in this work several sorts of sensors, namely GPS-based GNSS, a laser scanner, a camera, and a motion sensor, which have been commonly used for single intelligent vehicle operation. With these sensors, an intelligent vehicle can possess fairly complete perception abilities towards itself and the environment. We have reviewed the Bayesian filter framework that has been commonly used for recursive state estimation; we have also reviewed several recursive estimation methods that are derived from the Bayesian filter framework based on different kinds of approximations. We have discussed in detail the fundamental problems and the state-of-the-art methods concerning the cooperative localization, cooperative local mapping and moving objects detection. Based on these discussions, we propose a general architecture of cooperative localization using split covariance intersection filter (SCIF), an indirect vehicle-to-vehicle relative pose estimation method, and a new method for occupancy grid maps merging to handle the fundamental problems in cooperative localization, and cooperative local mapping and moving objects detection. We finally propose a brand

new idea of cooperative augmented reality which utilizes cooperative perception results to realize a special augmented effect.

We have provided a solution of multi-vehicles **c**ooperative **l**ocalization. We have reviewed the concept of estimate consistency and the SCIF. We have presented several forms of this filter together with their derivations and an original proof for the fusion consistency of this filter. We have introduced several basic functionalities as the condition for realizing cooperative localization; these functionalities are abstracted from field practice based on their feasibility in reality. We have described a general architecture of cooperative localization using the SCIF; as the architecture is decentralized, we have described from the perspective of an intelligent vehicle how it can evolve its state estimate using its motion measurements, how it can update its state estimate using its own absolute positioning measurements, and how it can update its state estimate with the data shared by neighbouring vehicles. We have presented the indirect vehicle-to-vehicle relative pose estimation strategy.

We have provided a solution of **c**ooperative local mapping and moving objects detection for laser scanner based intelligent vehicles. We have reviewed the method of occupancy grid based single vehicle local SLAM, including how to use laser scanner based range measurements to incrementally update the occupancy grid map estimate according to the inverse measurement model and how to estimate current vehicle local state (pose) with the last estimate of the vehicle local state and occupancy grid map. We have explained the different roles of vehicle local state and vehicle global state; we have described how vehicle local state estimate in SLAM can be used to assist vehicle global state estimation. We have presented the framework for occupancy grid maps fusion and merging by generalizing and formalizing its essential part into an optimization problem. We have proposed a new objective function that measures the consistency degree of maps alignment based on occupancy likelihood. We have adopted the spirit of genetic algorithms and designed a set of concrete procedures to search the optimal maps alignment. We have introduced the scheme of multi-vehicles cooperative moving objects detection based on occupancy grid maps merging; for a complete implementation, we have reviewed two basic moving objects detection methods, namely the consistency-based detection and the moving object map based detection.

We have extended the spirit of augmented reality to cooperative perception, forming the concept of *cooperative augmented reality* in the context of intelligent vehicle systems. We have specified the front-following vehicles scenario to which the proposed idea of cooperative augmented reality is applied. We have reviewed the pinhole camera model and described how to establish spatial relationship between two views (easily

extendable to multi-views case) according to perspective geometry. We have described several coordinates systems i.e. the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system that are concerned in an intelligent vehicle; we have introduced a technique of utilizing a 2D laser scanner to assist a mono-camera in estimating the visual perception depth approximately. We have presented how to map the visual perception of a vehicle onto that of another vehicle, abiding by the multi-views perspective geometry described. We have also introduced a new extrinsic calibration method for a camera and a 2D laser scanner, which can reveal all the spatial relationships among the camera's coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system, based only on the popular chessboard calibration practice with few extra measurements.

We have presented the experimental conditions and experimental results concerning cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality. We have presented the results of a simulation based comparative study which demonstrates the advantage of the proposed cooperative localization architecture using the split covariance intersection filter (the *SCIFCL* approach), especially for intelligent vehicles with heterogeneous absolute positioning ability. A prominent advantage of the SCIFCL method is that it enables good localization results to be naturally spread within a vehicle network in connection while always keeping a reasonable confidence for the state estimate of each vehicle. We have also presented the results of field tests (real-data) on cooperative localization, which lead to similar conclusions as in the simulation based comparative study. We have demonstrated the performance of the proposed occupancy grid maps merging method based on real-data tests. In spite of an intentionally exaggerated initial error range, local occupancy grid maps built by different vehicles can always be merged correctly using the proposed method; besides, the proposed occupancy grid maps merging method has the potential to recover the merging result from a *kidnapping* situation. We have demonstrated the performance of a proposed method coined as cooperative augmented reality, which realizes a vivid and lifelike effect of 'seeing' through the front vehicle for the following vehicle in a front-following vehicles scenario.

RÉSUMÉ

Le thème de recherche de cette thèse est la perception coopérative multi-véhicules appliquée au contexte des systèmes de véhicules intelligents. L'objectif général des travaux présentés dans cette thèse est de réaliser la perception coopérative de plusieurs véhicules (dite « *perception coopérative* »), visant ainsi à fournir des résultats de perception améliorés par rapport à la perception d'un seul véhicule (ou « *perception non-coopérative* »). Au lieu de concentrer nos recherches sur la performance absolue de la perception coopérative, nous nous concentrons sur les mécanismes généraux qui permettent la réalisation de la localisation coopérative et de la cartographie de l'environnement routier (y compris la détection des objets), considérant que la localisation et la cartographie sont les deux tâches les plus fondamentales pour un système de véhicule intelligent. Nous avons également exploité la possibilité d'explorer les techniques de la réalité augmentée, combinées aux fonctionnalités de perception coopérative. Nous baptisons alors cette approche « *réalité augmentée coopérative* ».
Par conséquent, nous pouvons d'ores et déjà annoncer trois contributions des travaux présentés: la localisation coopérative, la cartographie locale coopérative, et la réalité augmentée coopérative.

Description

Dans nos travaux, nous avons exploité plusieurs sortes de capteurs, à savoir un GNSS à base de GPS, un télémètre laser, une caméra, et des capteurs odométriques. Ces capteurs sont souvent employés pour le fonctionnement d'un véhicule intelligent et, grâce à ceux-ci, un véhicule intelligent est doté d'une capacité de perception assez complète lui permettant d'assurer sa propre localisation et la perception proprement dite de l'environnement.
Afin d'assurer la localisation du véhicule, une architecture à base de filtre Bayésien a été examinée ; celui-ci est couramment utilisé pour l'estimation d'état récursive. Ainsi, un rappel des diverses méthodes d'estimation récursives dérivées de l'architecture de filtre Bayésien est fait. Dans la suite, sont discutés en détail les problèmes fondamentaux et les méthodes existantes dans l'état-of-the-art concernant la localisation et la cartographie locale coopératives. D'après ces réflexions, nous proposons une architecture générale de localisation coopérative en utilisant le « split covariance

intersection filter » (ou SCIF), une méthode de l'estimation indirecte de la localisation relative Véhicule-à-Véhicule. De même, une nouvelle méthode de fusion de grilles d'occupation est présentée et ce, afin de traiter les problèmes fondamentaux en matière de la localisation coopérative et de cartographie locale coopérative.

Nous avons fourni une solution pour la localisation coopérative multi-véhicules. Nous avons rappelé le concept de consistance de l'estimation ainsi que le SCIF. Nous avons présenté plusieurs formes de ce filtre avec leurs dérivations et une preuve originale pour la consistance de la fusion de ce filtre. Nous avons introduit plusieurs fonctionnalités de base comme la condition pour réaliser la localisation coopérative. Nous avons introduit une architecture générale de localisation coopérative en utilisant un SCIF. Puisque l'architecture est décentralisée, nous avons décliné l'approche dans le cadre de la localisation d'un véhicule intelligent en s'appuyant sur ses capteurs de mouvement. Nous explicitons ainsi la manière dont il peut mettre à jour son estimation d'état en utilisant ses propres mesures de positionnement absolu, ainsi que la mise à jour de son estimation d'état avec les données partagées avec les véhicules voisins. Nous avons présenté la stratégie d'estimation indirecte de du positionnement relatif Véhicule-à-Véhicule.

Nous avons fourni une solution de cartographie locale coopérative pour les véhicules intelligents fondée sur la télémétrie laser. Nous avons décrit la méthode de SLAM local fondée sur la grille d'occupation. Nous faisons la distinction entre état local et état global puis nous décrivons comment les estimations de l'état local du véhicule obtenues par le SLAM peuvent être utilisées pour obtenir les estimations de l'état global de celui-ci. Nous avons présenté l'architecture de fusion des grilles d'occupation en formalisant le problème dans un cadre généralisé de problème d'optimisation. Nous avons proposé une nouvelle fonction objective qui mesure le degré cohérence de l'alignement des cartes fondé sur la probabilité d'occupation. Ensuite, nous avons proposé une approche fondée sur un algorithme génétique dans le but de rechercher l'alignement optimal des grilles. Nous avons enfin introduit l'architecture de la détection coopérative des objets en mouvement, fondé sur la fusion des grilles occupations. Pour une mise en œuvre complète, nous avons adopté deux méthodes de base pour la détection des objets en mouvement.

Nous avons exploité la notion de réalité augmentée à la perception coopérative, formalisant ainsi le concept de *« réalité augmentée coopérative »* appliquée au contexte des systèmes de véhicules intelligents. Nous nous sommes intéressés particulièrement au scénario de véhicules « leader-suiveur » auquel l'approche de réalité augmentée est appliquée. Pour cela, nous utilisons deux capteurs : un télémètre laser et une caméra.

Nous avons décrit comment établir une relation spatiale entre deux vues selon la géométrie perspective. Nous avons introduit une technique permettant à un télémètre laser 2D de fournir à une caméra des données lui permettant d'estimer la profondeur de perception visuelle. Nous avons présenté la façon de projeter la perception d'un véhicule sur celle d'un autre véhicule, en respectant la géométrie perspective décrite. Nous avons également introduit une nouvelle méthode de calibration extrinsèque pour une caméra et un télémètre laser 2D.

Nous avons présenté les conditions expérimentales et les résultats expérimentaux concernant la localisation coopérative, la cartographie locale coopérative et la réalité augmentée coopérative. Nous avons présenté les résultats d'une étude comparative fondée sur la simulation qui démontre l'avantage de l'architecture de localisation coopérative proposée utilisant le filtre SCIF (l'approche *SCIFCL*), notamment pour les véhicules intelligents avec des capacités de positionnement absolu hétérogènes. Un avantage important de la méthode SCIFCL est qu'elle assure une localisation améliorée naturellement répartie au sein du réseau de véhicules, tout en gardant une consistance raisonnable pour l'estimation de l'état de chaque véhicule. Nous avons également présenté les résultats de tests réels sur la localisation coopérative, qui conduisent à des conclusions similaires à l'étude comparative fondée sur la simulation. Nous avons démontré les performances de la méthode de fusion de grilles occupations, fondés sur des tests effectués avec des données réelles. En dépit d'une erreur initiale intentionnellement exagérée, les cartes locales construites par différents véhicules peuvent toujours être agrégées correctement en utilisant la méthode proposée. D'ailleurs, la méthode de fusion des grilles d'occupation a le potentiel de trouver une solution pour le problème dit de « kidnapping ». Nous avons démontré les performances de la méthode baptisée comme « *réalité augmentée coopérative* », qui réalise un effet vif de 'voir' à travers le véhicule leader pour le véhicule suiveur dans le scénario de véhicules « leader-suiveur ».

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 Introduction

Résumé

Le thème de recherche de cette thèse est la perception coopérative multi-véhicules appliquée au contexte des systèmes de véhicules intelligents. L'objectif général des travaux présentés dans cette thèse est de réaliser la perception coopérative de plusieurs véhicules (dite « *perception coopérative* »), visant ainsi à fournir des résultats de perception améliorés par rapport à la perception d'un seul véhicule (ou « *perception non-coopérative* »).

1.1 Context: Intelligent Vehicle Systems

Vehicles (automobiles, ground vehicles) have been praised as one of the greatest inventions in modern history, as they have revolutionized our living mode and have contributed enormously to the development of modern society. Thanks to them, the living range and the working range of human beings have been largely enhanced. For example, we can take an exciting job in one town while enjoying a desirable habitation in another town.

Since several decades ago, researchers have been making efforts on incorporating various intelligent functions into traditional vehicles, with a goal of realizing *intelligent vehicle systems* that make driving experience more convenient, more efficient, and safer. Take some scenarios as examples, as follows:

Imagine that you are driving to Paris for the first time and you intend to go to a given destination whose location you do not know yet. In this case, an intelligent vehicle localization system [Skog & Handel 2009] can estimate current location of the vehicle, match the location onto a pre-stored digital map, and computes an itinerary to guide you from your current location to your destination. The localization system will update the estimate of vehicle location in real-time and always keep you on the correct itinerary.

During your trip, you are a bit distracted by the exotic architectures on road sides and have not paid attention to gradual deviation of your vehicle; then an intelligent vision system that performs automatic lane detection [Li & Nashashibi 2011b] can monitor the vehicle-lane relative position and signal a lane departure warning message in time. You might have not paid attention to traffic lights or traffic speed signs either, then the vision system that performs traffic light detection [Charrette & Nashashibi 2009] and traffic speed sign detection [Moutarde *et al.* 2007] can also signal relevant message in time and display the traffic information.

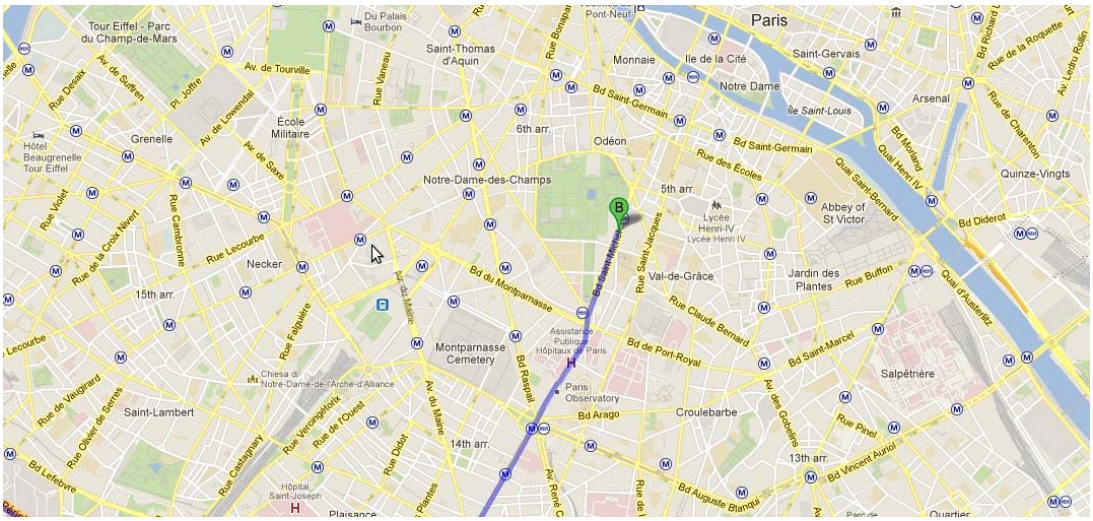


Figure 1.1 Vehicle navigation guidance with a digital map (modification on pictures from Google maps)

You might get tired of adjusting the vehicle speed; then an Adaptive Cruise Control (ACC) system [Vahidi & Eskandarian 2003] will liberate your foot from oil-pedal control and adjust the vehicle speed automatically. Suddenly, a pedestrian comes out from no where and rush across the road; for this sudden event, a system of pedestrian detection [Gate *et al.* 2009] will have rapid detection and recognition of the pedestrian and send proper commands to the vehicle controllers to avoid collision with the pedestrian.

After a long driving, you finally arrive at your destination, yet with a fatigued body. At this moment, an automatic vehicle parking system [Xu *et al.* 2000] will take charge of vehicle parking and save you from all these last steps of vehicle maneuvering.

Besides the examples listed above, an ocean of research works on intelligent vehicle systems can be found in literature. The research context of this dissertation is also the field of intelligent vehicle systems. In next sub-sections, we will specify our research focus in the context of **intelligent vehicle systems (IVS)**.

1.2 Vehicles Cooperation

Most research works in the field of intelligent vehicle systems focus on **SINGLE vehicle operation**, i.e. the intelligent vehicle performs environment perception, decision making, and action execution, based only on its own sensor information and its own planning, without interacting with other intelligent vehicles.

A typical example is vision based autonomous navigation [Thorpe *et al.* 1988] [Pomerleau 1989], where the on-vehicle vision processing module (based on either

mono-camera or stereo-camera) process digital image data to generate vehicle control law directly or generate useful spatial information of the environment that can be used to guide the behavior of the vehicle. Another typical example is laser scanner based *simultaneous localization and mapping* (SLAM) [Wang *et al.* 2003], the vehicle establishes spatially consistent relationship among its sequence of observations, generates a consistent environment representation (the process of *mapping*), and localizes itself with respect to this environment representation (the process of *localization*). Moreover, other examples of single intelligent vehicle operation can be found in the applications of GPS-based vehicle localization and navigation [Kao 1991] [Abuhadrous *et al.* 2003], pedestrian detection [Enzweiler & Gavrila 2009], vehicle detection [Sun *et al.* 2006], general objects detection [Bertozzi & Broggi 1998] [Labayrade *et al.* 2005], and vehicle lateral control and vehicle longitudinal control [Rajamani 2005] etc.

The application background of intelligent vehicle systems is the outdoor traffic environment; a noticeable feature of outdoor traffic environment is that thousands of vehicles operate in the same environment. We can make a fair analogy between the outdoor traffic environment and our society, where an intelligent vehicle can be compared to an individual human being.

Each of us possesses the ability of sensing the environment and the ability of reacting to the environment; one has the potential to survive by one's own ability, as how Robinson Crusoe did on a remote tropical island. In a society, however, we always cooperate with each other, instead of being totally independent. For example, when we arrive at a new place and want to search a certain street, we tend to consult some local passers-by for a quick access to our destination; without this cooperation, we might spend hours on searching the destination and suddenly find ourselves back again to a place we have passed by. When we want to enter into a cinema, we form a queue based on certain rule and pass the entrance orderly; without this cooperation, we would bump into the entrance randomly and might get stuck into a stalemate at the entrance.

In short, the cooperation among people in our society makes our lives more convenient and more efficient. Similarly, the cooperation among intelligent vehicle systems in traffic environment would also bring convenience and efficiency to traffic users.

Early motivation for performing vehicles cooperation lies in the idea of increasing infrastructure capacity via cooperative platooning (or cooperative adaptive cruise control) [Raza & Ioannou 1996] [Tsugawa *et al.* 2000] [Tsugawa *et al.* 2001] [Bruin *et al.* 2004]. For a highway segment, its capacity is limited by the safety interval distance between neighboring vehicles; the smaller this distance is, the larger the highway

capacity is. In cooperative platooning, down-stream vehicle can share its motion state and motion intention with upstream vehicles which can then take anticipatory actions and avoid jerk movements; as a result, the gap between neighboring vehicles can be reduced while string stable behaviors can be maintained, as illustrated in Figure 1.2. In other words, compared with non-cooperative platooning, cooperative platooning requires smaller safety distance, which can enhance highway capacity because vehicles can pass the highway segment more tightly.

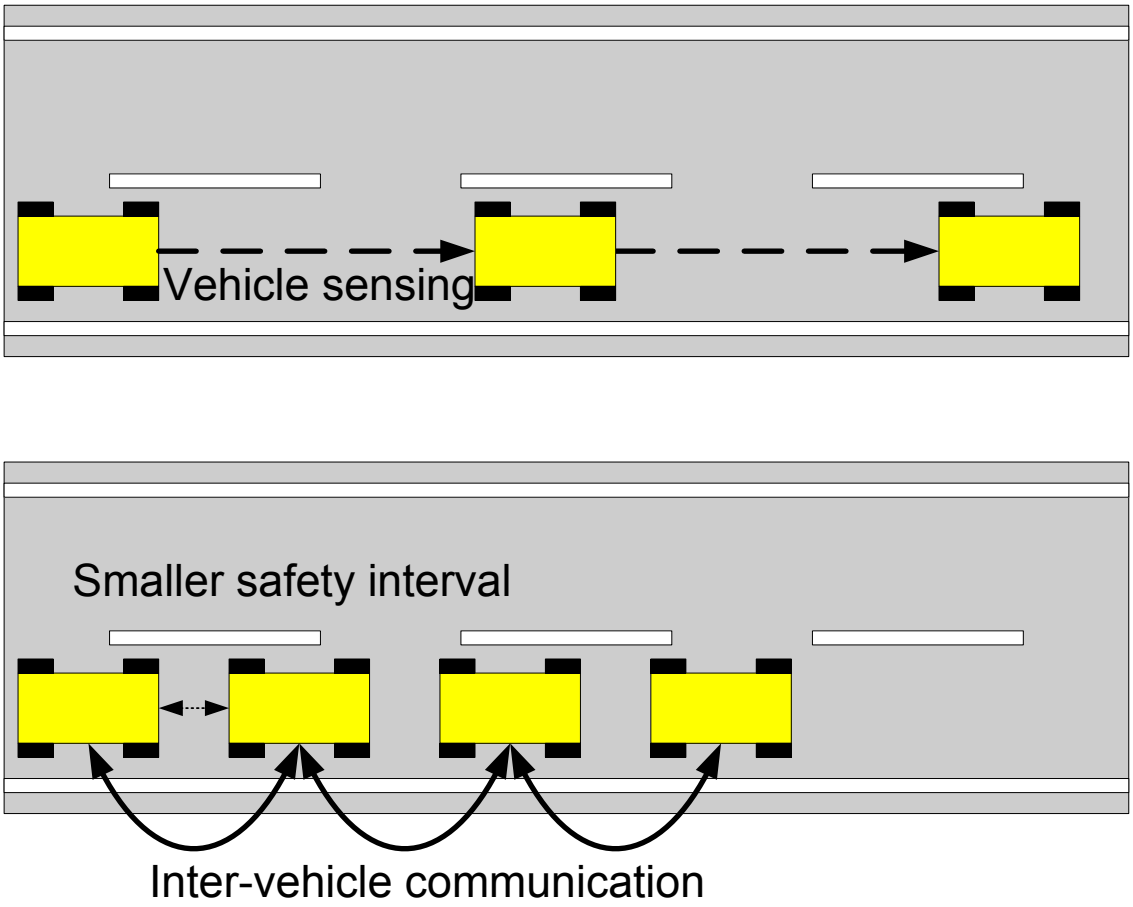


Figure 1.2 Vehicles platooning: (top) non-cooperative platooning; (bottom) cooperative platooning

Another motivation for performing vehicles cooperation stems from the need to guarantee navigation safety, which stimulates the development of cooperative collision warning-avoiding systems (CCWAS) [Li & Wang 2006] [Farahmand & Mili 2009] [Tan & Huang 2006]. Take an intersection scenario as an example, as illustrated in Figure 1.3. Vehicle A and Vehicle B move toward an intersection; they have several possible motion modes at the intersection. Some motion mode might result in a collision

accident if the drivers of the two vehicles misjudge the situation. With inter-vehicle communication, they can share the information of their position, motion state, and motion intention etc; then they can evaluate the possibility of collision. If a risk of collision exists, they can take proper actions in time to prevent collision from happening. Besides intersection scenarios [Chan & Bougler 2005] [Li & Wang 2006], cooperative collision warning-avoiding systems are also valuable for guaranteeing navigation safety at lane changing scenarios [Ammoun *et al.* 2007] [Li *et al.* 2005].

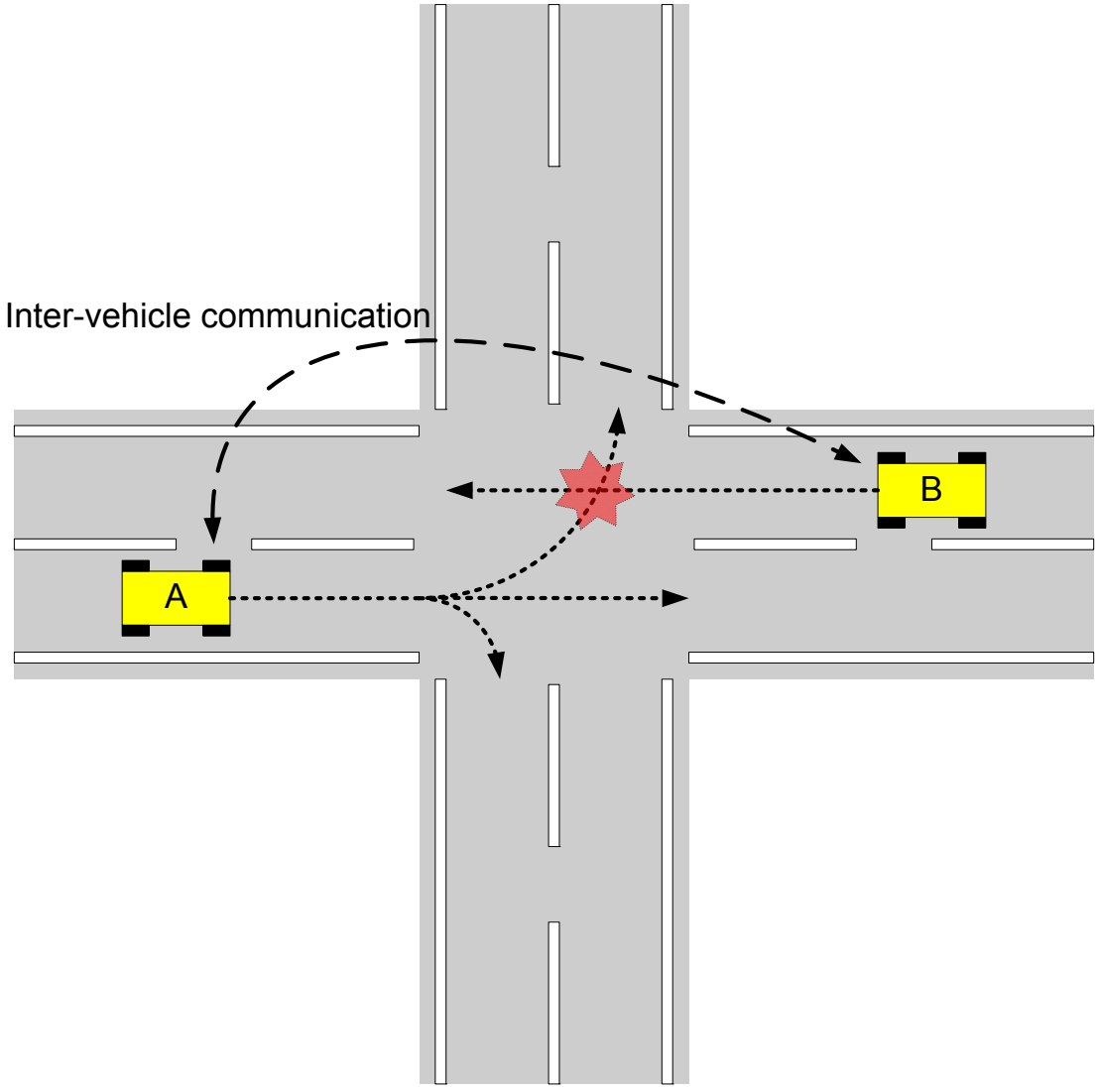


Figure 1.3 Cooperative collision warning-avoiding at an intersection

Moreover, the philosophy of vehicles cooperation has been applied to diverse kinds of scenarios, providing possible solutions for special demands of the society. For example, a new type of elderly driver assistance systems [Tsugawa *et al.* 2007] enables a host vehicle (driven by an assisting driver) to assist or escort the guest vehicle (driven by an elderly driver) through inter-vehicle communication. Europe has supported several large projects such as I-WAY [Rusconi *et al.* 2007], CVIS [Koenders & Vreeswijk 2008], and COOPERS [Piao & McDonald 2008], which attempt to integrate state-of-the-art V2V and V2I communication technologies and cooperative philosophy into comprehensive traffic scenarios.

1.3 Cooperative Perception

Generally speaking, single vehicle operation concerns two functionalities: *perception* and *control*. The vehicle perceives the environment and its own state, and then determines how to react to the environment; the vehicle repeatedly executes the procedures of perception and control, as illustrated in Figure 1.4.

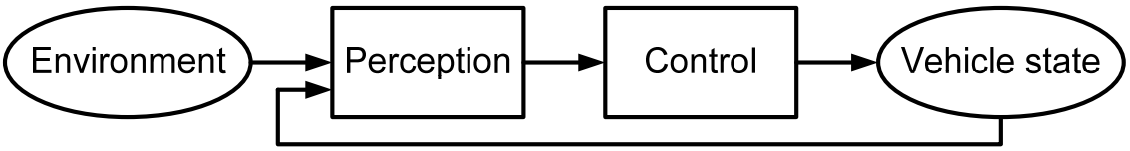


Figure 1.4 Intelligent vehicle operation paradigm

Vehicles cooperation also concerns the functionalities of perception and control by each individual vehicle. Moreover, the vehicles in cooperation can share and fuse their perceptions, which results in the functionality of **cooperative perception**; they can also share their motion intentions and coordinate their actions, which results in the functionality of **cooperative control**. Besides cooperative perception and cooperative control, vehicles cooperation also concerns the functionality of **vehicular communication** which plays a fundamental role for vehicles cooperation, as illustrated in Figure 1.5.

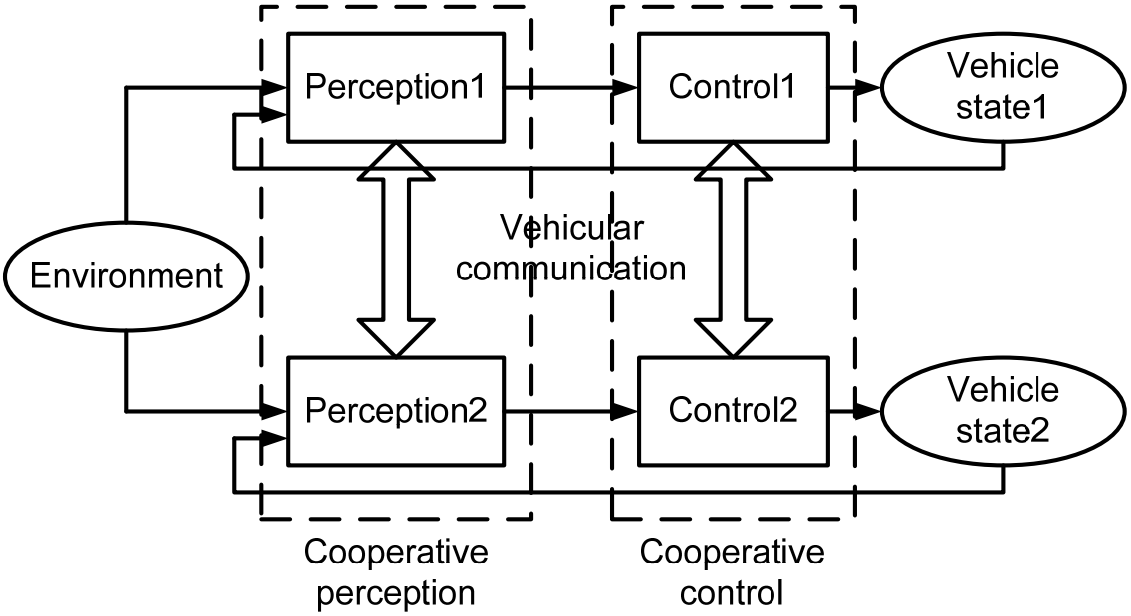


Figure 1.5 Vehicles cooperation paradigm

This dissertation will focus on the functionality of **cooperative perception**; see the hierarchy of research areas in Figure 1.6. In the next sub-section, we will explain the value of cooperative perception in the context of intelligent vehicle systems and specify the problems and applications of cooperative perception concerned in this dissertation.

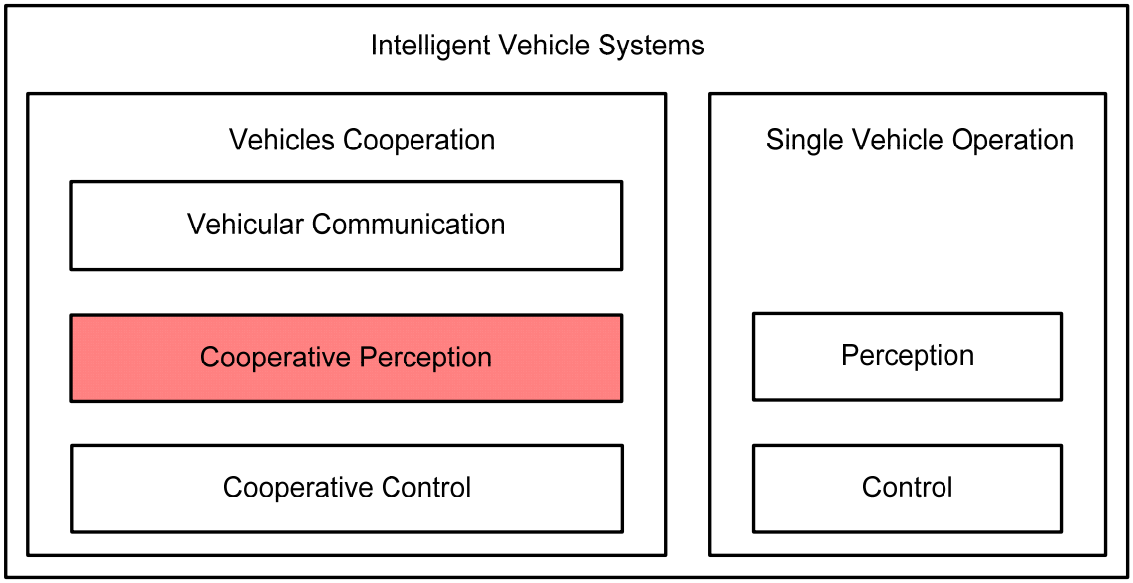


Figure 1.6 Hierarchy of research areas

1.3.1 Motivation

As mentioned above, most research works in the field of intelligent vehicle systems focus on single vehicle operation; naturally, most research works on vehicle perception focus on the perception performed independently by a single vehicle.

Compared with single vehicle perception (or non-cooperative perception), the motivation for developing cooperative perception can be illustrated by a typical traffic scenario, i.e. an overtaking scenario as shown in Figure 1.7. A vehicle is overtaking another vehicle while the overtaken vehicle (the first vehicle) occludes the view of the overtaking vehicle (the second vehicle). This scenario is potentially dangerous; for example, in the case where a careless pedestrian is rushing across the road in front of the first vehicle, then what might happen?

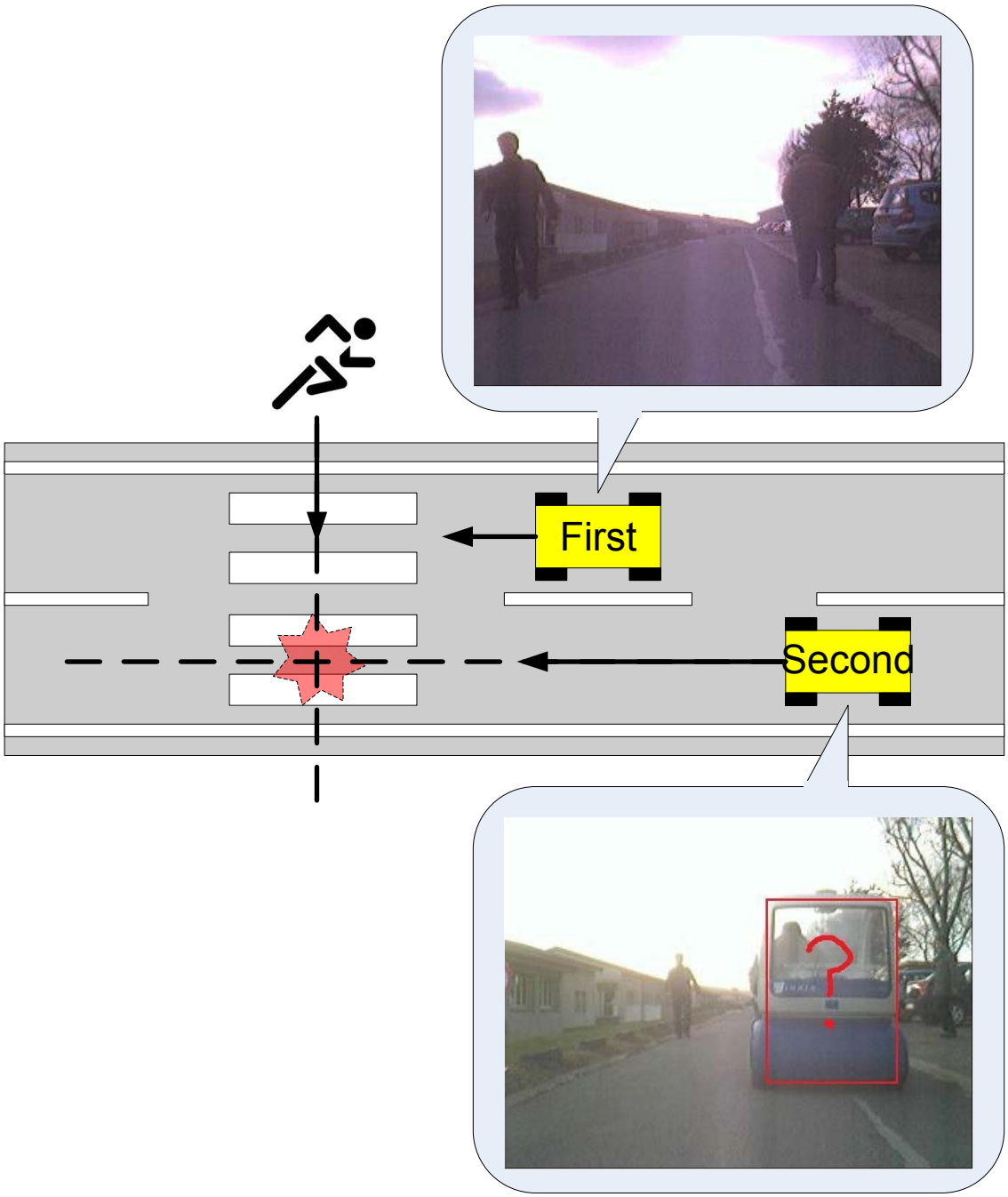


Figure 1.7 Overtaking scenario: potentially dangerous

For reasons of safety and efficiency, the second vehicle always wants to know: what objects are there occluded by the front vehicle? If there are some objects with risky trajectories, the second vehicle can make certain anticipatory actions such as deceleration or even braking-down. On the other hand, if there is no object, the second

vehicle can just keep going with high speed, this will be more efficient for the second vehicle and possibly other vehicles following the second vehicle.

Unfortunately, the second vehicle can never answer this question by itself, simply because it cannot perceive the occluded environment. No matter how good sensors and how good algorithms the second vehicle is using, it can not have any inference about the occluded environment. Then, a simple motivation for developing cooperative perception is to help the second vehicle answer this question. More specifically, the basic idea of cooperative perception is to let the first vehicle share its perception with the second vehicle, in order that the second vehicle can "perceive" the occluded environment.

Another traffic scenario that demonstrates the value of cooperative perception is illustrated in Figure 1.8(top). Two vehicles A and B navigate in the same area; each of them has self-localization ability; they can also estimate the relative pose between them using their perception components.

Concerning the localization for vehicle B (similar reasoning can be carried out for vehicle A), the position of vehicle B can be estimated by vehicle B itself (shown in Figure 1.8(middle)). Besides, the position of vehicle B can be estimated indirectly from the perspective of vehicle A, as vehicle A can compound the estimate of relative pose between the two vehicles and the estimate of its own position (shown in Figure 1.8(bottom)).

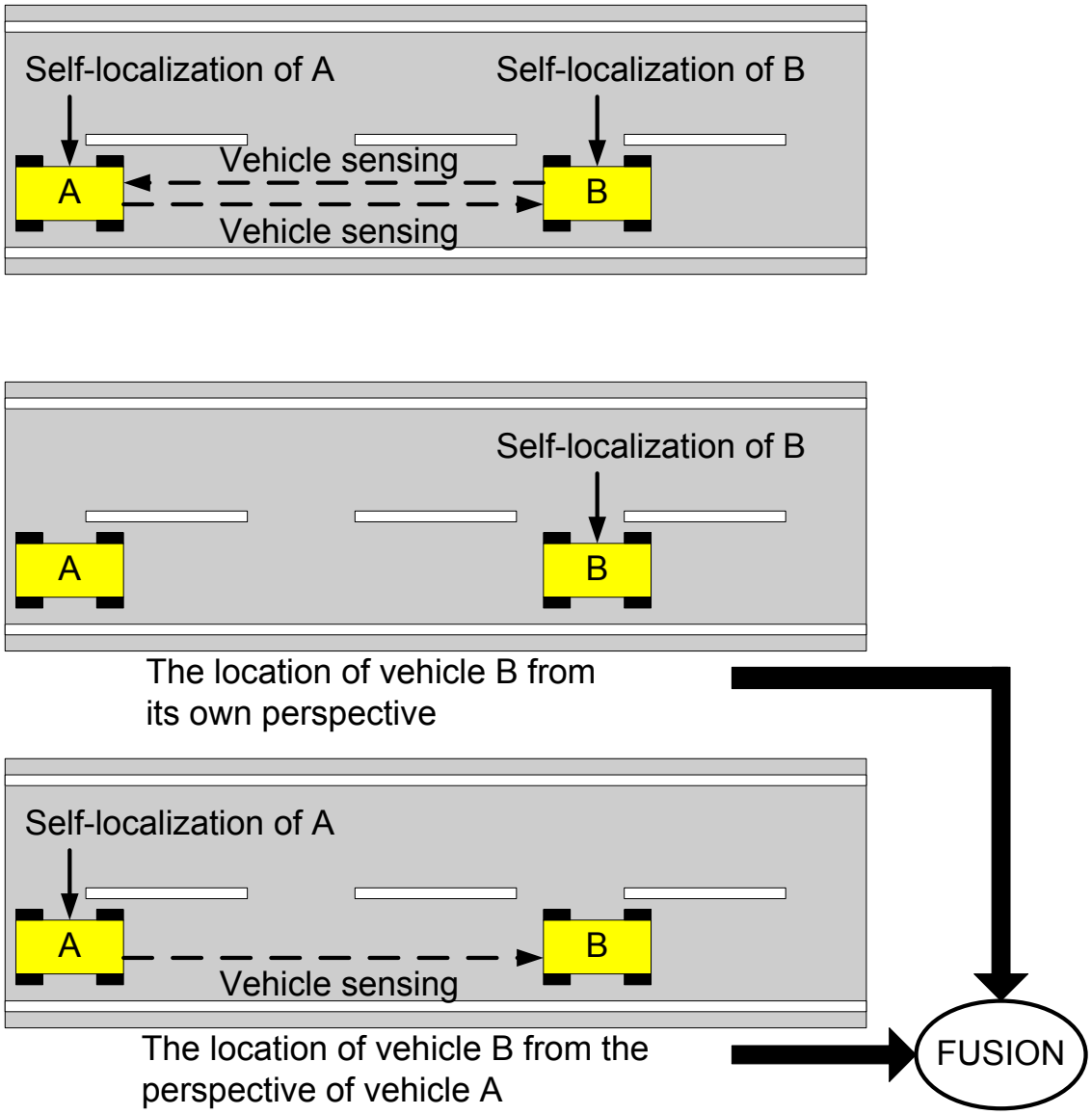


Figure 1.8 Multi-vehicles cooperative localization

In other words, vehicle B can be localized based on two sources of data: one is of its own; the other one is of vehicle A. Better localization results might be achieved for vehicle B, if we fuse the two sources of data. The advantage of this fusion is especially noticeable for heterogeneous systems: imagine that vehicle A has high-quality positioning configuration whereas vehicle B has low-quality positioning configuration, then the high-quality positioning result of vehicle A can significantly improve the positioning results of vehicle B.

1.3.2 Problem statement

The general methodology of the presented works in this dissertation is to **realize multi-intelligent vehicles cooperative perception, which aims at providing better vehicle perception result compared with single vehicle perception (or non-cooperative perception)**.

Before further explanations for this methodology, we would like to review several underlying tasks of single vehicle perception. The essential purpose of vehicle perception is to provide relevant information, based on which the vehicle can decide how to react to the environment. For this purpose, first, the vehicle has to know its spatial state (position and orientation) with respect to a global reference, or to a local reference, or to both; the process of estimating this spatial state of the vehicle is usually referred to as **vehicle localization** (or **localization** for short). Second, the vehicle also has to establish spatial representation for the environment objects. Some environment objects are stationary, for example, buildings, road infrastructures etc; the process of establishing spatial representation for the stationary objects is usually referred to as **mapping**. Some environment objects are moving (or dynamic), for example, moving vulnerable road users; the process of establishing spatial representation for the moving objects is usually referred to as **moving objects detection**. The process of establishing spatial representation for both stationary objects and moving objects in the environment is referred to as **mapping and moving objects detection**. All these underlying tasks are important for successful and safe navigation of the vehicle.

When we extend single vehicle perception to multi-vehicles cooperative perception, we will naturally focus on the underlying issues of **cooperative localization** and **cooperative mapping and moving objects detection** for cooperative perception. Simply speaking, cooperative localization is a process where multiple vehicles perform localization cooperative, i.e. a vehicle can utilize the data of other vehicles to assist the localization of the vehicle itself. Similarly, cooperative mapping and moving objects detection is a process where multiple vehicles perform mapping and moving objects detection cooperatively, i.e. a vehicle can utilize the data of other vehicles to assist its tasks of mapping and moving objects detection.

It is worthy noting that the role of these cooperative perception functionalities is **optional** instead of being **mandatory**. A vehicle can choose to cooperate with other vehicles or not, depending on its judgement whether it can benefit from others or whether it can help others. For example, a vehicle with high quality GPS (Global Positioning System) and high quality IMU (Inertial Measurement Unit) might trust largely in its own positioning results and chooses not to fuse the positioning information

of other vehicles. It is like when we visit a new country, we tend to consult local people about our destinations (as we think local people are likely to be familiar with the local environment); if we find ourselves only with other foreigners nearby, we might choose to rely on ourselves to find our destinations (as we think these foreigners are unlikely to be familiar with the local environment).

Since the role of cooperative localization and cooperative mapping and moving objects detection is optional, then a question arises naturally: how to determine whether to perform these cooperative functions or not? However, we will not discuss about this issue in this dissertation, as this issue will deviate the research focus of cooperative perception. What we will tackle is: **how to realize cooperative localization** and **how to realize cooperative mapping and moving objects detection**?

It is also worthy noting that cooperation is NOT omnipotent and we had better NOT expect that cooperative perception can unlimitedly make improvements over single vehicle perception. For example, we can not expect that two vehicles both with ten-meter level positioning accuracy can achieve centimeter-level accuracy through cooperative localization. The performance of cooperative perception largely depends on the perception ability of each individual vehicle in cooperation. The better the single vehicle perception is, the better the cooperative perception tends to be.

Therefore, instead of focusing our research works on the absolute performance of cooperative perception, **we would rather focus on the general mechanisms which enable the realization of above mentioned cooperative perception functionalities based on commonly used sensor configurations; we would rather examine the advantages of these cooperative perception functionalities relative to single vehicle perception.**

The final goal for developing intelligent vehicle systems in the long run is that all the vehicles in our society can reliably operate in full automated mode (or with only few human interventions such as designating the destination of the vehicle users). However, there is still considerable gap between current technical ability and the ability to achieve above goal. Before this final goal becomes true, it would be desirable that current techniques can be adapted for driver assistance. Better visualization of vehicle perception results would better assist the driver to judge the environment. **Augmented reality** techniques can make the perception visualization more direct and more vivid, which are commonly used in driver assistance oriented applications. Concerning cooperative perception, we will study the possibility of **taking advantage of cooperative perception to generate augmented reality effects that would be valuable for drivers**, which we name as **cooperative augmented reality**.

1.4 Contributions

The contributions of the presented works in this dissertation consist in three aspects: cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality.

1.4.1 Cooperative Localization

We provide a solution of multi-vehicles **c**ooperative **l**ocalization (CL). First, we make an abstraction of the basic functionalities that are commonly available in the context of intelligent vehicle systems. Based on these abstracted functionalities, we propose a general architecture of cooperative localization using split covariance intersection filter. Second, concerning the functionality of vehicle-to-vehicle relative pose estimation that is fundamental for realizing cooperative localization, we propose a new method i.e. *the indirect vehicle-to-vehicle relative pose estimation method* to perform this functionality. We carry out a simulation based comparative study among the proposed cooperative localization architecture and several reference methods. Besides simulation, we also carry out cooperative localization in reality and present the results of real-data tests.

1.4.2 Cooperative Local Mapping and Moving Objects Detection

We provide a solution of **c**ooperative **lo**cal **m**apping and **m**oving **o**bjects **d**etection for laser scanner based intelligent vehicles. The method architecture is as follows: each vehicle establishes in real-time a local occupancy grid map and performs moving objects detection based on the established occupancy grid map. During vehicles cooperation, the local occupancy grid maps of different vehicles are merged, so that these different vehicles can be spatially related to each other; then the moving objects detection results of these vehicles can also be merged. As part of this method architecture, a new method for occupancy grid maps merging is proposed, which consists in a new objective function that measures the consistency degree of maps alignment and a genetic algorithm that searches for the optimal maps alignment.
We carry out real-data tests on the proposed method and demonstrate its performance.

1.4.3 Cooperative Augmented Reality

We will extend the spirit of augmented reality to cooperative perception, forming the concept of ***c****ooperative* ***a****ugmented* ***r****eality* (CAR) in the context of intelligent vehicle systems. We provide a solution of cooperative augmented reality, which integrates the techniques of cooperative local mapping and augmented reality to generate an

augmented reality effect of ‘seeing through’ front vehicle. As part of the provided solution, a new method is proposed for extrinsic co-calibration of a camera and a 2D laser scanner.
We will demonstrate the ‘seeing through’ effect of the introduced cooperative augmented reality method, based on real-data tests.

1.5 Thesis Outline

This dissertation is organized as follows: In chapter 2, we review in details the problems and the state-of-the-art methods concerned in the cooperative perception issues discussed in this dissertation. From chapter 3 to chapter 5, we respectively introduce our solutions of cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality. In chapter 6, we describe concrete implementation and integration of the proposed methods on our experimental vehicle platforms and demonstrate experimental results. In chapter 7, we summarize the works presented in this dissertation and discuss about their future extensions.

CHAPTER 2 Cooperative Perception – State-of-the-Art

Résumé

Dans nos travaux, nous avons exploité plusieurs sortes de capteurs, à savoir un GNSS à base de GPS, un télémètre laser, une caméra, et des capteurs odométriques. Ces capteurs sont souvent employés pour le fonctionnement d'un véhicule intelligent et, grâce à ceux-ci, un véhicule intelligent est doté d'une capacité de perception assez complète lui permettant d'assurer sa propre localisation et la perception proprement dite de l'environnement. Afin d'assurer la localisation du véhicule, une architecture à base de filtre Bayésien a été examinée ; celui-ci est couramment utilisé pour l'estimation d'état récursive. Ainsi, un rappel des diverses méthodes d'estimation récursives dérivées de l'architecture de filtre Bayésien est fait. Dans la suite, sont discutés en détail les problèmes fondamentaux et les méthodes existantes dans l'état-of-the-art concernant la localisation et la cartographie locale coopératives. D'après ces réflexions, nous proposons une architecture générale de localisation coopérative en utilisant le « split covariance intersection filter » (ou SCIF), une méthode de l'estimation indirecte de la localisation relative Véhicule-à-Véhicule. De même, une nouvelle méthode de fusion de grilles d'occupation est présentée et ce, afin de traiter les problèmes fondamentaux en matière de la localisation coopérative et de cartographie locale coopérative.

2.1 Introduction

As we have stated in previous chapter: the performance of cooperative perception largely depends on the perception ability of each individual vehicle in cooperation; instead of focusing our research works on the absolute performance of cooperative perception, **we would rather focus on the general mechanisms which enable the realization of previously mentioned cooperative perception functionalities based on common used sensor configurations; we would rather examine the advantages of these cooperative perception functionalities relative to single vehicle perception.**

Therefore, before going into detailed discussion on the cooperative perception issues, we had better first give an introduction of the state-of-the-art sensor configurations that are common for single intelligent vehicle operation. This can provide a baseline for our research works: First, concerning methodology, we can judge whether the proposed methods are generally applicable to vehicles with common sensor configurations? Second, concerning performance, we can examine what benefits can cooperative perception bring to vehicles with common sensor configurations?

The execution of a perception task can be basically treated as a process of estimating certain state of interest (vehicle pose, environment map etc) based on the measurements of certain vehicle sensors. Following the introduction of intelligent vehicle sensor configurations, we would like to introduce some general mathematical foundations of estimation theory, which have been widely applied in real-time vehicle perception tasks. Then, we study in details the problems and the state-of-the-art methods concerned respectively in the issues of cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality, which have been briefly introduced in previous chapter.

2.2 Intelligent Vehicle Sensor Configurations

We review several kinds of sensors that are commonly used in nowadays intelligent vehicles.

2.2.1 Global Positioning System (GPS)

The Global Positioning System (GPS) is part of a satellite-based navigation system developed by the United States Department of Defense under its NAVSTAR satellite program. More detailed description of the history and technologies of the GPS can be found in an ocean of literature on this topic; refer to [Grewal *et al.* 2001] [Farrell &

Barth 1998] for examples. Briefly speaking, a GPS receiver (with related processing units) can provide information of its location and the GPS universal time, via analysis of the signals transmitted by the GPS satellites whose orbital information is known.

Several factors account for the popularity of the GPS in the applications of intelligent vehicle systems: First, it can provide positioning measurement with respect to a **global** reference (a reference fixed with the environment where the intelligent vehicles operate). Second, it can provide the global positioning measurement **directly**, without the need for certain extra conditions such as pre-registration works about the operation environment. Third, it can provide **error-bounded** positioning measurement, without suffering from accumulated positioning errors.

Although the U.S. America was the first to develop satellites-based global positioning system, yet nowadays it is not the only provider of satellites-based global positioning services. Some other entities such as Russia, Europe, and China have also been developing global positioning systems (in different names); for example, the GLObal Navigation Satellite System (GLONASS) by Russia, the GALILEO positioning system by European Union, and the COMPASS (or Bei Dou in Chinese) navigation system by China. Therefore, in this dissertation, the referring of the terms *global positioning system (GPS)* does not necessarily indicate that specific system developed by U.S.A, but generally indicates any possible system which performs the functionality of global positioning based on satellites.

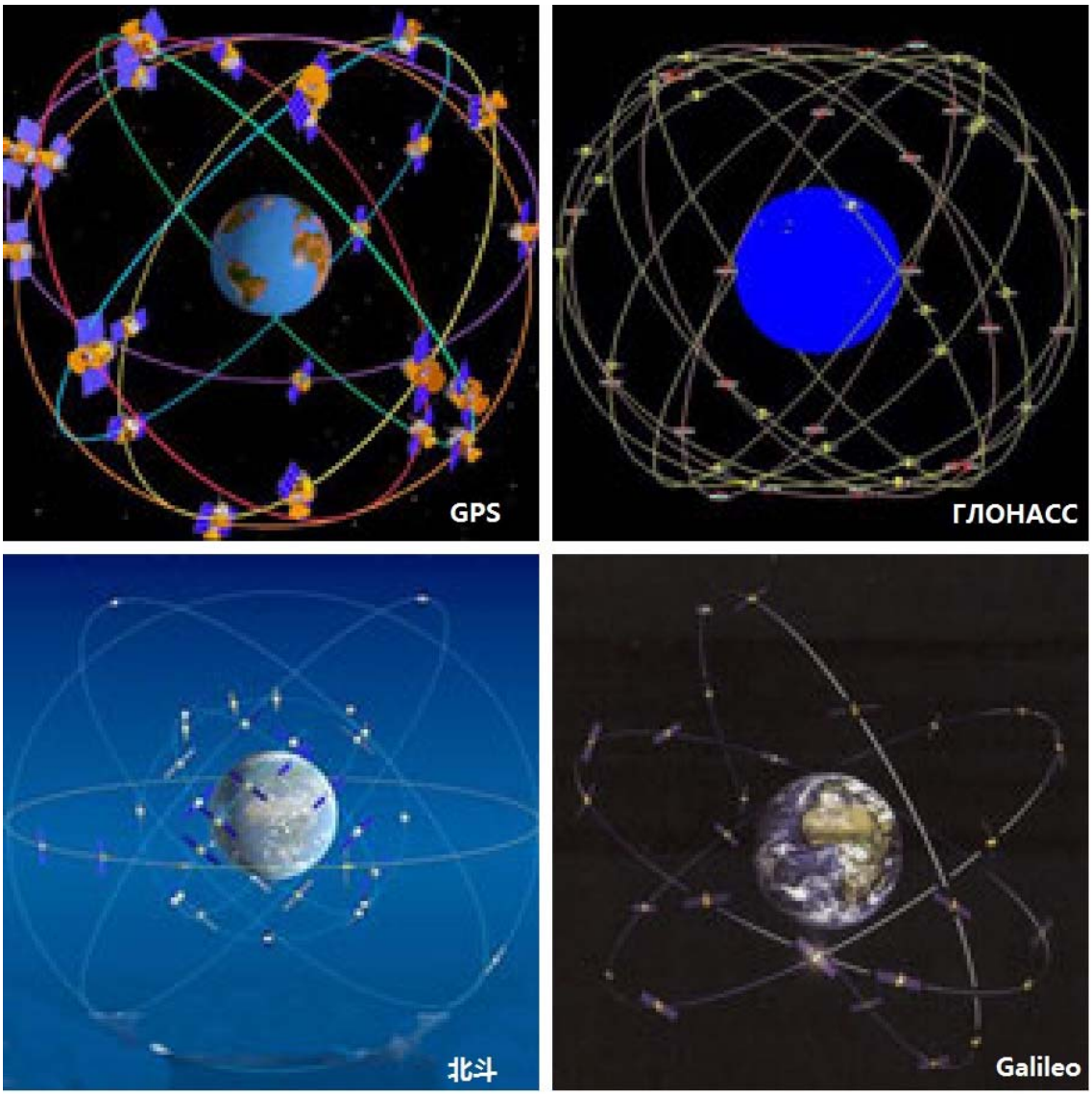


Figure 2.1 Satellites-based global positioning systems (modification on pictures from Huan Qiu Shi Bao i.e. Global Times)

2.2.2 Laser Scanner

Laser scanner, or in other names such as *laser rangefinder*, *laser telemeter*, is a kind of device that can measure its distance to environment objects by emitting laser beams, receiving the reflection of the laser beams, and computing the distance traversed by the laser beams.

A laser scanner can rapidly provide reliable range measurements with fairly small range errors (centimeter level errors). In other words, a laser scanner enables a vehicle to

efficiently monitor the spatial relationship between the vehicle and the surrounding objects. As a result, laser scanner plays an important role in guaranteeing vehicle navigation safety, especially for the purpose of environment objects collision avoidance (avoiding collision onto vulnerable road users, other vehicles, environment infrastructure etc).

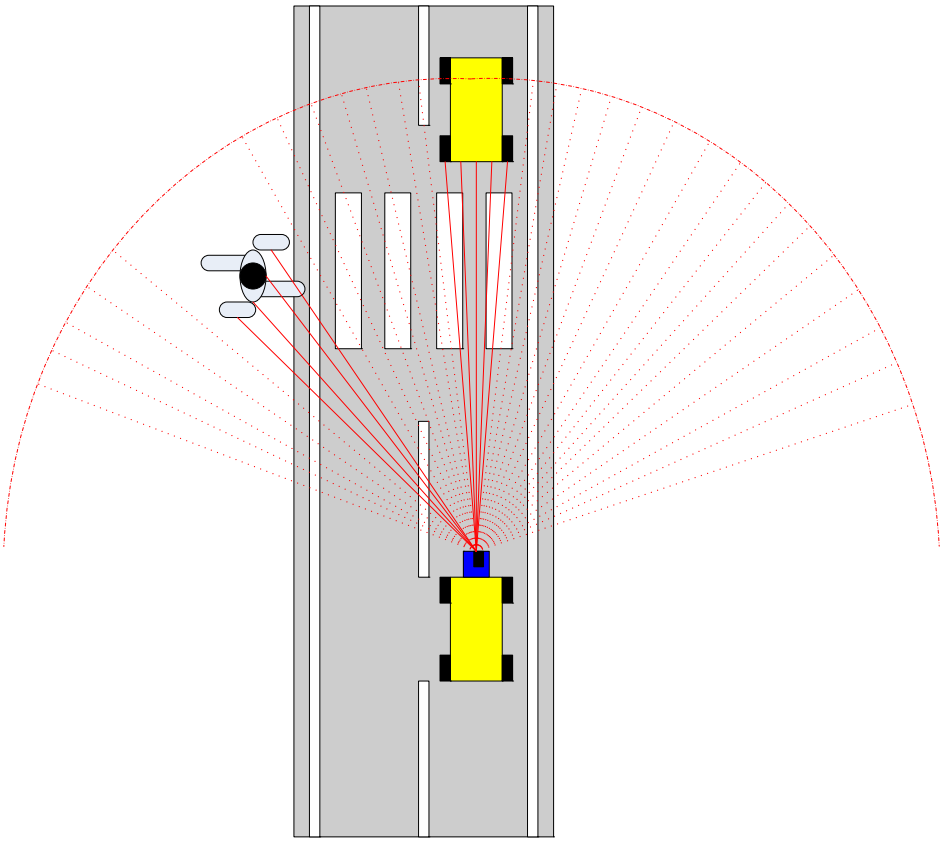


Figure 2.2 Range measurements provided by a laser scanner

2.2.3 Camera

Cameras can provide a kind of sensing data that is related to the most important perception system of human beings, i.e. our vision. As our vision provides the most part of information for our reasoning about the environment, cameras can also provide a large amount of information for intelligent vehicle systems to make inference about the environment. For example, based on vision data, on-vehicle vision systems can perform tasks of lane marks detection and traffic signs detection that are almost impossible for other kinds of perceptive sensors such as laser scanners. Vision data provide plenty of

clues for automatic recognition of environment objects (pedestrians, vehicles etc). Besides, camera data (original or with certain visualization effects) can be easily comprehended by human beings, which makes cameras valuable for the purpose of driver assistance, human-machine interaction etc.

2.2.4 Motion Sensor

Motion sensors are a kind of proprioceptive sensors (in contrast with exteroceptive sensors such as cameras and laser scanners) that are usually equipped on an intelligent vehicle to monitor vehicle motion state (longitudinal motion and lateral motion). Motion sensors used for intelligent vehicles include odometers (including steering encoders), accelerometers, gyroscopes etc.

Normally, motion sensors can output motion measurements at comparatively high frequency (for example, 100 Hz). Therefore, motion measurements can be used to predict vehicle state when other sorts of measurements are temporarily unavailable. Besides, motion measurements can be fused with other sorts of measurements to enhance the accuracy of vehicle state estimates.

2.2.5 Integration

We have briefly introduced several sorts of sensors, namely GPS, laser scanner, camera, and motion sensor. For certain specific application, we might be able to resort to only one sort of these sensors. For example, for lane detection and lane following, we only need a vision system.

On the other hand, it has been a tendency to incorporate all these sensors into an intelligent vehicle, in order that the vehicle possesses fairly complete perception ability towards itself and the environment. The reasons are two-folds:

First, the functionality of each sort of these sensors is irreplaceable by the others. Without GPS, the vehicle has no error-bounded inference about its global position. Without camera, the vehicle has no visual data for computer-based image processing or for human-oriented visualization. Without laser scanner, the vehicle can not have range data of high reliability and accuracy. Without motion sensor, the vehicle can not have direct monitoring of its motion state.

Second, the availability of all these sensors can complement each other and facilitate the functioning of each other. In other words, they are mutually beneficial. For example, motion data can be used to facilitate processing of range data and vision data; they can be fused with global positioning measurements. Range data based processing results can be used to correct motion data.

An example of intelligent vehicle equipped with GPS, laser scanner, camera, and motion sensors is illustrated in Figure 2.3.

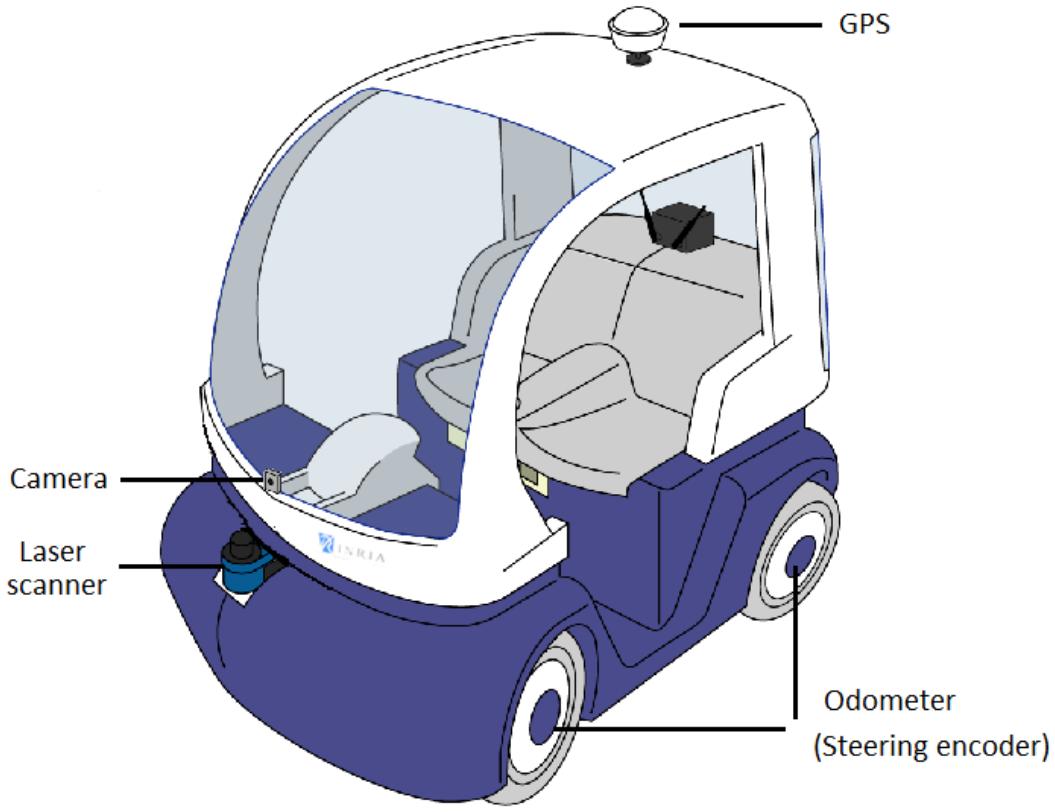


Figure 2.3 Intelligent vehicle sensor configurations (CyCab vehicle platform)

2.3 Recursive State Estimation

2.3.1 State, State Estimation, and Perception

The term *state* is frequently adopted in a wide range of domains; the definition of state depends on concrete research and application areas. In the context of intelligent vehicle systems, we could generally think state as the collection of all properties of the vehicles and the environment. In reality, however, we can not really deal with all the properties, because they are infinite. In fact, we do not need to deal with all the properties; for example, we do not need to care about that an intelligent vehicle consists of how many molecules. In other words, we need to selectively deal with partial properties of the vehicles and the environment that are important in certain sense; then we treat these

partial properties as *state*. As in this dissertation, we deal with the *spatial properties* of the vehicles and the environment and treat them as state.

State estimation addresses the problem of estimating the state of interest (the properties we care about) via analysis of sensor data. The need for state estimation lies in two aspects of reasons: First, certain state elements might not be directly observable from sensor data; they can only be inferred from sensor data. For example, GPS only outputs position measurement for the vehicle, whereas vehicle orientation can only be inferred from the process of vehicle pose estimation i.e. vehicle localization. Second, for those state elements directly observable from sensor data, techniques such as filtering and data fusion in state estimation can reduce the uncertainty of the raw sensor data of these state elements.

After a brief introduction of *state* and *state estimation*, now we could further specify the meaning of *perception*. Literally, *perception* means 1) the ability of perceiving (sensing, recognizing etc), 2) the process of perceiving, and 3) the result of perceiving. In this dissertation, the *perception*, when used in general manner, still conveys all these senses of meaning. For example, *cooperative perception* implies the ability of perceiving cooperatively, the process of perceiving cooperatively, and the results of perceiving cooperatively.

Concerning the third sense of meaning i.e. the result of perceiving, the *perception* only implies **current** result of perceiving, as we focus on real-time vehicle perception. In other words, the *perception* means **current state estimate** that might be obtained based (directly and indirectly) on analysis and fusion of a temporal sequence of sensor measurements. For example, when we express "a vehicle shares its perception with another vehicle" or "associate the perceptions of two vehicles", here, the *perception* has such sense of meaning i.e. the current state estimate maintained by a vehicle.

2.3.2 Vehicle-Environment Interaction

During the operation of an intelligent vehicle, it gets sensor measurements about itself and the environment. We can assign its sensor measurements into two categories: vehicle proprioceptive measurements and vehicle exteroceptive measurements. The terms proprioceptive and exteroceptive have been briefly mentioned in the introduction of motion sensors; here we would like to explain their difference.

Vehicle proprioceptive measurements: the *proprio* means "of one's own"; vehicle proprioceptive measurements are measurements that only concern the vehicle itself. Vehicle motion measurements belong to such kind of measurements.

As will be explained later, among the four kinds of sensor measurements (GPS, laser scanner, camera, and motion sensor) introduced previously, only vehicle motion measurements are proprioceptive for the vehicle. Thus in this dissertation, the concepts of vehicle proprioceptive measurements and vehicle motion measurements are not distinguished from each other and they are used interchangeably.
We generally denote vehicle motion measurements as **u**, and use a subscript to denote time. The vehicle motion measurements at time *t* will be denoted as

$$\mathbf{u}_t$$

The notation

$$\mathbf{u}_{t_1:t_2} = \mathbf{u}_{t_1}, \mathbf{u}_{t_1+1}, ..., \mathbf{u}_{t_2}$$

denotes the set of vehicle motion measurements from time t_1 to time t_2, for $t_1<t_2$.
Besides being proprioceptive, another feature for vehicle motion measurements is that they are directly related to vehicle control actions that cause vehicle state to change. Therefore, the denotation **u** bears two aspects of indications: first, it denotes the passive measurements on vehicle motion; second, it denotes the motion actions that actively contribute to vehicle state transition.

Vehicle exteroceptive measurements: the *extero* means "exterior"; vehicle exteroceptive measurements are measurements that concern the vehicle and the environment. GPS measurements, laser scanner measurements, and camera measurements belong to such kind of measurements.
Sometimes, GPS measurements are regarded as proprioceptive measurements; however, we would rather treat GPS measurements as exteroceptive measurements, because what GPS measures is not some properties of the vehicle itself, but the spatial relationship between the vehicle and a group of satellites.
We could do an imaginary experiment to better understand the difference between proprioceptive and exteroceptive: imagine that a cover is always around the vehicle and it absolutely cuts the vehicle away from environment. In this situation, GPS, laser scanner, and camera will lose their functioning, because environment objects outside the vehicle become totally "invisible"; so these sensors are exteroceptive. In contrast, motion sensors can still function; so vehicle motion sensors are proprioceptive.
We generally denote vehicle exteroceptive measurements as **z**; and

$\mathbf{z}_t$ (for time t)

$\mathbf{z}_{t_1:t_2} = \mathbf{z}_{t_1}, \mathbf{z}_{t_1+1}, ..., \mathbf{z}_{t_2}$ (from time t_1 to time t_2, for $t_1<t_2$)

Vehicle-environment state: currently, we do not distinguish between the vehicles and the environment; we generally denote their state (i.e. their properties that we want to estimate) together as

$\mathbf{x}_t$ (for time t)

$\mathbf{x}_{t_1:t_2} = \mathbf{x}_{t_1}, \mathbf{x}_{t_1+1}, ..., \mathbf{x}_{t_2}$ (from time t_1 to time t_2, for $t_1<t_2$)

The purpose of vehicle perception is to estimate the state **x** based on vehicle exteroceptive measurements **z** and vehicle motion measurments **u**, as illustrated in Figure 2.4.

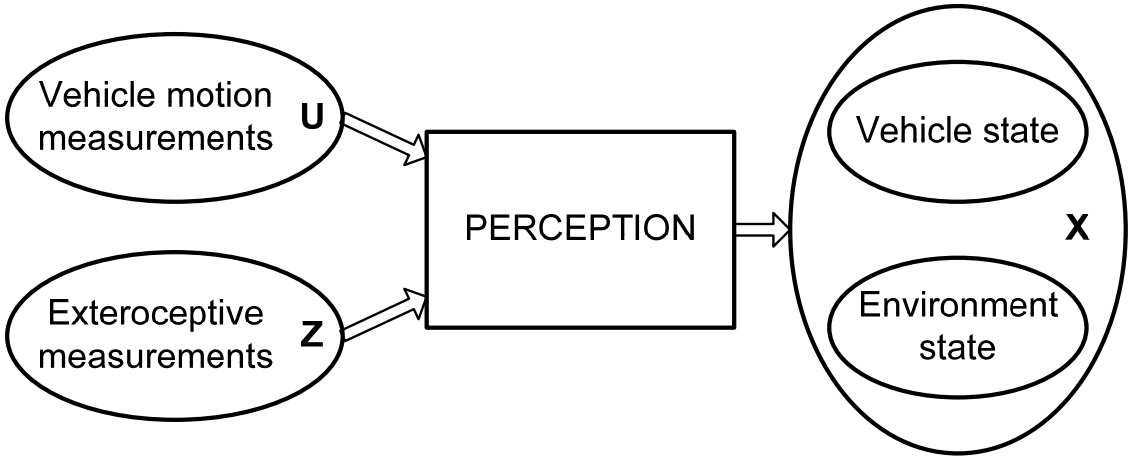


Figure 2.4 General perception process: State estimation

In probabilistic terms, the problem is to estimate the posterior belief distribution of the following form:

$bel(\mathbf{x}_{1:t}) = p(\mathbf{x}_{1:t} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t})$

Since we mainly focus on real-time perception that cares about state estimate at current time, the problem of vehicle perception turns to estimate the posterior distribution as specified in (2-1):

$$bel(\mathbf{x}_t) = p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \tag{2-1}$$

2.3.3 Recursive Estimation: Bayesian Filter Framework

In reality, the operation of an intelligent vehicle is a dynamic process of interaction between the vehicle and the environment; the vehicle has to perform perception tasks in dynamic way: the vehicle has to re-estimate the state of interest from time to time, because both the vehicle and the environment are changeable.

Vehicle measurements accumulate as time passes; for each round of state estimation, we can not expect that the estimation is based on all historical measurements, because the huge amount of measurements will make the problem intractable neither at computational level nor at storage level. A more reasonable practice is to base a new round of state estimation on the results of last round of state estimation and the new measurements since last round of state estimation, in short words, to perform state estimation in **recursive** way.

We can derive a recursive formulism of state estimation from (2-1), using some fair assumptions. Via Bayes rule we have

$$\begin{aligned} p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) &= \frac{p(\mathbf{z}_t \mid \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t})}{p(\mathbf{z}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t})} \\ &= \delta \; p(\mathbf{z}_t \mid \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \end{aligned} \tag{2-2}$$

Where δ is a normalization factor. The first term of the right side in above equation, denotes the (conditional) probability of current exteroceptive measurement $\mathbf{z}_t$ conditioned on the state $\mathbf{x}_t$, past exteroceptive measurements $\mathbf{z}_{1:t\text{-}1}$, and vehicle motion measurements $\mathbf{u}_{1:t}$. We can follow the **Markov assumption** or the **complete state assumption** [Thrun *et al.* 2005] i.e. the knowledge of $\mathbf{x}_t$ is sufficient to predict $\mathbf{z}_t$; no past exteroceptive measurements or vehicle motion measurements would provide additional information about $\mathbf{z}_t$. In mathematical terms, $\mathbf{z}_t$ is conditionally independent of $\mathbf{z}_{1:t\text{-}1}$ and $\mathbf{u}_{1:t}$:

$$p(\mathbf{z}_t \mid \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) = p(\mathbf{z}_t \mid \mathbf{x}_t) \tag{2-3}$$

Using the theorem of total probability on the second term of the right side in (2-2) and we have:

$$\begin{aligned} p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) &= \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\ &= \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1} \end{aligned} \tag{2-4}$$

During the derivation of (2-4), the Markov assumption is used another time, i.e. $\mathbf{x}_t$ is only dependent of $\mathbf{x}_{t-1}$ and $\mathbf{u}_t$, whereas past exteroceptive measurements and past vehicle motion measurements convey no information on the state $\mathbf{x}_t$.
Substitute (2-3) and (2-4) into (2-2):

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) = \delta \; p(\mathbf{z}_t \mid \mathbf{x}_t) \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1} \tag{2-5}$$

Equation (2-5) describes how we can update the old distribution $p(\mathbf{x}_{t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1})$ with new measurements $\mathbf{u}_t$ and $\mathbf{z}_t$. This recursive estimation process is the so-called **Bayesian filter** in literature, which can be illustrated by **dynamic Bayesian network** (DBN) [Murphy 2002], as in Figure 2.5.

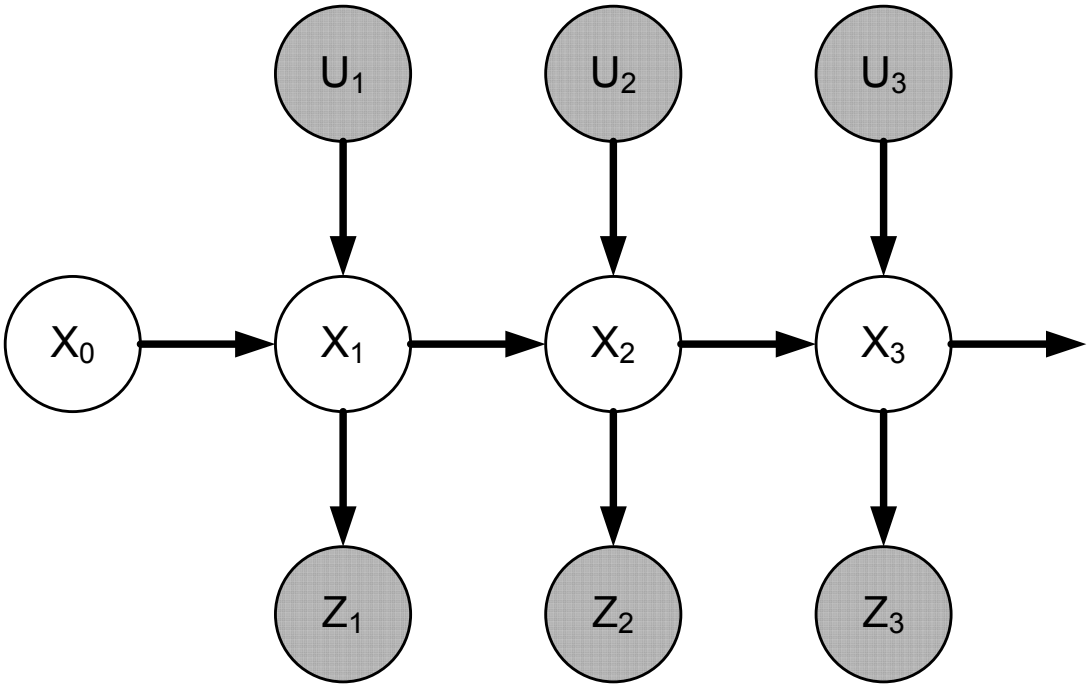


Figure 2.5 Dynamic Bayesian network: a framework for recursive state estimation

In real implementation, we can not strictly follow the mathematical framework of Bayesian filter given by (2-5), because it is normally numerically intractable. We have to make certain approximations on (2-5) to simplify the original Bayesian filter into tractable formulism.

Next, we are going to introduce several recursive estimation methods that are derived from the Bayesian filter (2-5) based on different kinds of approximations.

2.3.4 Kalman Filter

The Kalman filter was proposed by Rudolph Emil Kalman [Kalman 1960], as a technique of state filtering and prediction for linear systems.

Given a linear state transition model (or system evolution model) $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t)$ as

$$\mathbf{x}_t = A_t \mathbf{x}_{t-1} + B_t \mathbf{u}_t + \boldsymbol{\varepsilon}_t$$

and a linear measurement model $p(\mathbf{z}_t \mid \mathbf{x}_t)$ as

$$\mathbf{z}_t = C_t \mathbf{x}_t + \boldsymbol{\nu}_t$$

If we follow **Gaussian** noise assumption:

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) = \frac{1}{\sqrt{2\pi R_t}} \exp\{-\frac{1}{2}(\mathbf{x}_t - A_t\mathbf{x}_{t-1} - B_t\mathbf{u}_t)^T R_t^{-1}(\mathbf{x}_t - A_t\mathbf{x}_{t-1} - B_t\mathbf{u}_t)\}$$

and

$$p(\mathbf{z}_t \mid \mathbf{x}_t) = \frac{1}{\sqrt{2\pi Q_t}} \exp\{-\frac{1}{2}(\mathbf{z}_t - C_t\mathbf{x}_t)^T Q_t^{-1}(\mathbf{z}_t - C_t\mathbf{x}_t)\}$$

where R_t and Q_t are respectively the covariance of the zero-mean random variables $\boldsymbol{\varepsilon}_t$ and $\boldsymbol{\upsilon}_t$.

Then the Bayesian filter (2-5) will become the **Kalman filter** (KF) as:

$$\begin{aligned}
\overline{\mathbf{x}}_t &= A_t\hat{\mathbf{x}}_{t-1} + B_t\mathbf{u}_t \\
\overline{\boldsymbol{\Sigma}}_t &= A_t\boldsymbol{\Sigma}_{t-1}A_t^T + R_t \\
K_t &= \overline{\boldsymbol{\Sigma}}_t C_t^T (C_t\overline{\boldsymbol{\Sigma}}_t C_t^T + Q_t)^{-1} \\
\hat{\mathbf{x}}_t &= \overline{\mathbf{x}}_t + K_t(\mathbf{z}_t - C_t\overline{\mathbf{x}}_t) \\
\boldsymbol{\Sigma}_t &= (I - K_t C_t)\overline{\boldsymbol{\Sigma}}_t
\end{aligned}$$

The estimated belief distribution also follows the Gaussian distribution

$$bel(\mathbf{x}_t) = \frac{1}{\sqrt{2\pi\boldsymbol{\Sigma}_t}} \exp\{-\frac{1}{2}(\mathbf{x}_t - \hat{\mathbf{x}}_t)\boldsymbol{\Sigma}_t^{-1}(\mathbf{x}_t - \hat{\mathbf{x}}_t)\}$$

Detailed derivation of the Kalman filter can be referred to [Kalman 1960] [Thrun *et al.* 2005] [Grewal & Andrews 2000].
For nonlinear state transition model and measurement model

$$\begin{aligned}
\mathbf{x}_t &= g(\mathbf{x}_{t-1}, \mathbf{u}_t) + \boldsymbol{\varepsilon}_t \\
\mathbf{z}_t &= h(\mathbf{x}_t) + \boldsymbol{\upsilon}_t
\end{aligned}$$

We can linearize the nonlinear functions with first-order Taylor expansion and also follow the Gaussian noise assumption. Then we get a set of estimation procedures similar to KF, which is referred to as **Extended Kalman Filter** (EKF) in literature.

$$\bar{\mathbf{x}}_t = g(\hat{\mathbf{x}}_{t-1}, \mathbf{u}_t)$$
$$\bar{\boldsymbol{\Sigma}}_t = G_t \boldsymbol{\Sigma}_{t-1} G_t^T + R_t$$
$$K_t = \bar{\boldsymbol{\Sigma}}_t H_t^T (H_t \bar{\boldsymbol{\Sigma}}_t H_t^T + Q_t)^{-1}$$
$$\hat{\mathbf{x}}_t = \bar{\mathbf{x}}_t + K_t(\mathbf{z}_t - h(\bar{\mathbf{x}}_t))$$
$$\boldsymbol{\Sigma}_t = (I - K_t H_t)\bar{\boldsymbol{\Sigma}}_t$$

The G_t is the Jacobian matrix of function g with respect to $\mathbf{x}_{t-1}$; the H_t is the Jacobian matrix of function h with respect to $\mathbf{x}_t$.
The KF (or EKF) has been favored in a wide range of applications, as it provides us a convenient and efficient way to approximate full posterior distribution of the state by estimating only a mean vector and a covariance matrix. Although the Gaussian assumption on which the KF (or EKF) is based can not be strictly satisfied in reality, yet the KF (or EKF) normally gives desirable estimation results for unimodal estimation problem (for example, for fusing GPS measurements and vehicle motion measurements both of which are unimodal).

2.3.5 Incremental Maximum Likelihood Estimation

During recursive state estimation, the KF (or EKF) tries to maintain an estimate of full posterior distribution of the state; however, this practice will soon become intractable as the dimension of the state increases to hundreds or even more.
In contrast, if we do not maintain an estimate of full posterior distribution of the state, but only keeps the most likely state value (with maximum likelihood), then the Bayesian filter framework (2-5) will become the incremental maximum likelihood estimation framework:

$$\hat{\mathbf{x}}_t = \arg\max_{\mathbf{x}_t} \{ p(\mathbf{z}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \hat{\mathbf{x}}_{t-1}, \mathbf{u}_t) \}$$

The advantage of the incremental maximum likelihood estimation framework lies in its simplicity, as it does not need to maintain a full posterior distribution estimate. The incremental maximum likelihood estimation framework is especially suitable for

estimating large dimensional state, such as in the process of simultaneous localization and mapping (SLAM) [Gutmann & Knolige 1999] [Lu & Milios 1997a] [Vu 2009].

2.3.6 Sampling-based Method: Particle Filter

In KF, the Gaussian distribution assumption makes an analytical (closed-form) expression available for (2-5). If no specific distribution modeling is adopted, then (2-5) can only be executed with numerical discretization. The discrete form of the Bayesian filter is expressed in (2-6) (where the *k* and *i* denote the discretization index).

$$p(\mathbf{x}_{k,t} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) = \delta\ p(\mathbf{z}_t \mid \mathbf{x}_{k,t}) \sum_i p(\mathbf{x}_{k,t} \mid \mathbf{x}_{i,t-1}, \mathbf{u}_t) p(\mathbf{x}_{i,t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) \tag{2-6}$$

A naive implement of the discrete Bayesian filter (2-6) might be computationally expensive or even intractable. As a result, sampling-based techniques have been utilized to reduce the computational complexity of (2-6); a popular sampling-based method is particle filter, or sequential Monte Carlo method, which tries to approximate the state distribution with a group of particles and corresponding particle weights. Instead of updating the belief over all state space, the particle filter only updates the particles and their weights, thus largely reducing computational burden compared with naïve implementation of the discrete Bayesian filter.

One merit of the particle filter lies in the ability of its particles and corresponding weights to generally represent arbitrary state distribution (especially for multi-modal distribution). The more the particles are, the better the representation ability is, and the better the particle filter performs. On the other hand, the particle filter is not computationally efficient compared with the KF (or EKF) for dealing with estimation problem of the same state dimension. Besides, as the state dimension increases, the number of particles usually has to increase exponentially in order to keep the representation ability of the particles, which will further cause explosion of computational complexity. One can refer to [Doucet *et al.* 2000] [Doucet *et al.* 2001] for further knowledge on particle filter.

2.4 Cooperative Localization

2.4.1 Operation Architecture

For single vehicle localization, there is usually only one fusion center (the on-vehicle processing unit) which collects data from on-vehicle sensors (such as GPS [Redmill *et al.* 2001] [Rezaei & Sengupta 2007], cameras [Thorpe *et al.* 1988], laser scanners

[Wang *et al.* 2003] [Lingemann *et al.* 2005], or hybrid sorts of sensors [Li *et al.* 2010]), processes these sensor data, and outputs estimate of vehicle pose. In other words, the operation architecture for single vehicle localization is usually **centralized**.

Concerning multi-vehicles cooperative localization, if the size of a group of vehicles in cooperation is small, then the centralized architecture [Roumeliotis2002] [Howard *et al.* 2002] might be a possible solution. In the centralized architecture, one fusion center collects data from all the vehicles in cooperation, computes a global state estimate and distribute the global state estimate among all the vehicles, as illustrated in Figure 2.6. The fusion center can be situated at an infrastructure site or one of the vehicles in cooperation.

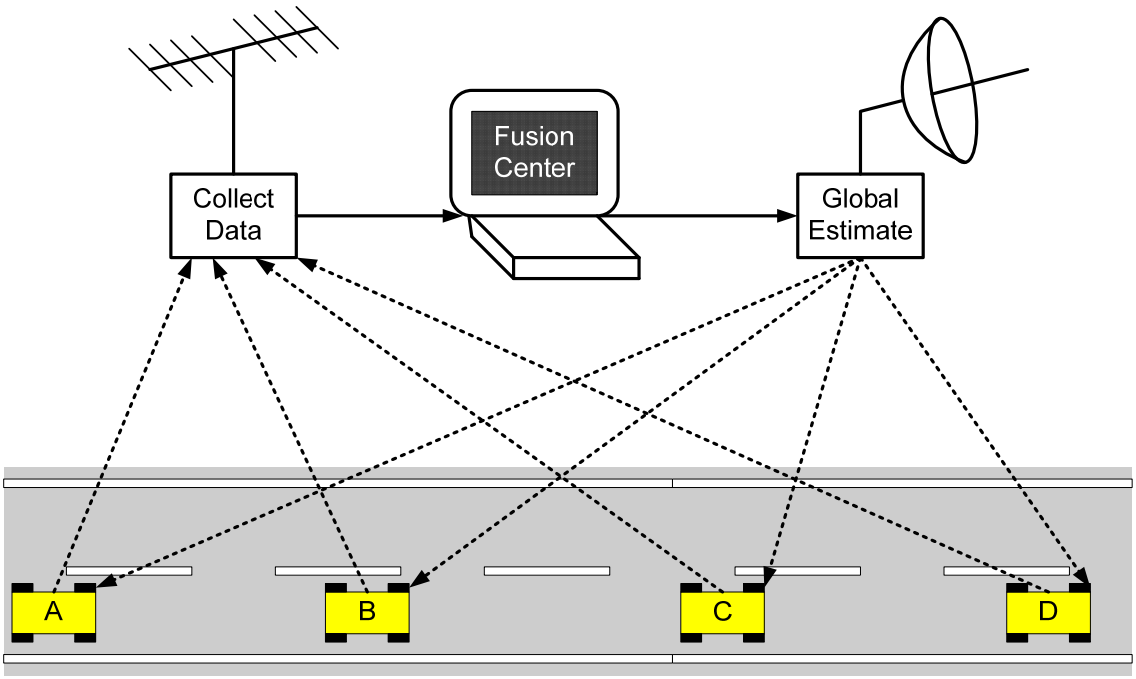


Figure 2.6 Centralized operation architecture

The advantage of centralized architecture is its convenience for data processing, as in single vehicle localization; all the data collected for the fusion are independent and there is no issue of data correlation which will be explained later. However, the centralized architecture suffers from large communication burden and delay; besides, as the number of vehicles increases, the computational burden will increase much faster. Moreover, the centralized architecture is inflexible in highly dynamic vehicle networks which are always the case in real traffic environment.

Instead of a centralized architecture, **a decentralized architecture**, where multiple fusion centers exist and each of them handles only local information, turns out to be a desirable solution in the long run. In the decentralized (or distributed) architecture, each vehicle collects data from its local surrounding environment and fuses the collected data for itself. The advantage of decentralized architecture lies in its flexibility for dealing with dynamic vehicle networks and comparatively low computational burden for each fusion center.

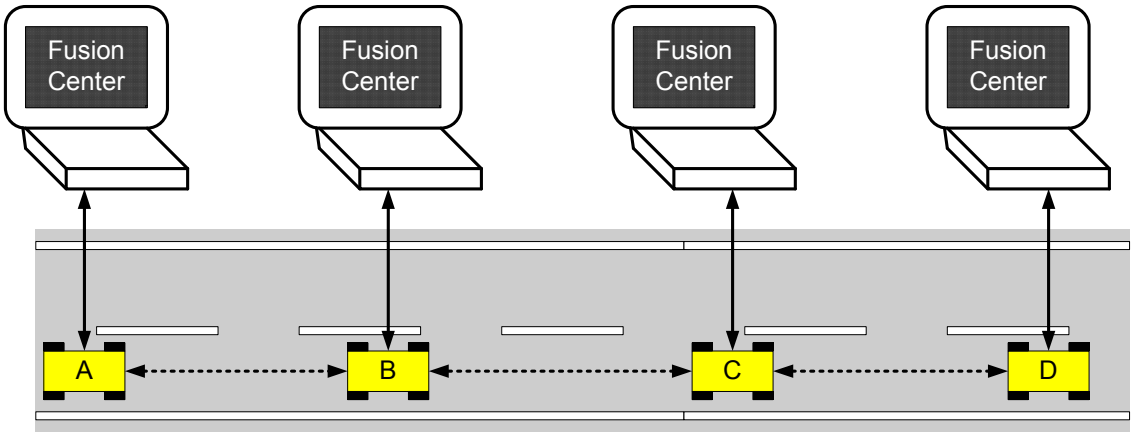


Figure 2.7 Decentralized (or distributed) operation architecture

2.4.2 How to Handle Inter-Estimates Correlation?

An important issue arises for decentralized architecture, i.e. how to handle inter-estimates correlation, i.e. the correlation (interdependency) among different estimate sources. Careless handling of this correlation will lead to circular reasoning [Howard *et al.* 2003] which further leads to the over-convergence problem, i.e. the estimates quickly converge to inaccurate values or even severely diverged values while extremely large confidence is given to these inaccurate values.

A simple example of the circular reasoning and over-convergence is illustrated in Figure 2.8. Here we have three vehicles vehicle 1 (V_1), vehicle 2 (V_2), and vehicle 3 (V_3); the uncertainty degree of their position estimates is indicated by the ellipses around them. Suppose at one moment, vehicle 1 gets a new positioning measurement; the uncertainty degree of this new measurement is also indicated by an ellipse. This new measurement can help vehicle 1 to reduce the uncertainty degree of its position estimate. With better self-positioning result, vehicle 1 can also have better estimation of the position of vehicle 2, which can reduce the position estimate uncertainty of vehicle 2. In

similar way, the position estimate uncertainty of vehicle 3 can be reduced with the help of vehicle 2. Then vehicle 3 might share its updated estimate with vehicle 1, which again can reduce the position estimate uncertainty of vehicle 1; so on and son on, such kind of circular update can continues until the position estimates of all the three vehicles converge to certain over-confident results.

The estimates of these vehicles might be inaccurate; as they have already established over-confidence on the inaccurate estimates, they can not correct these inaccurate estimates in time even if some new measurements which bear brand new information are available. As this unreasonable circular reasoning continues, the estimates might diverge far and far away from the truth.

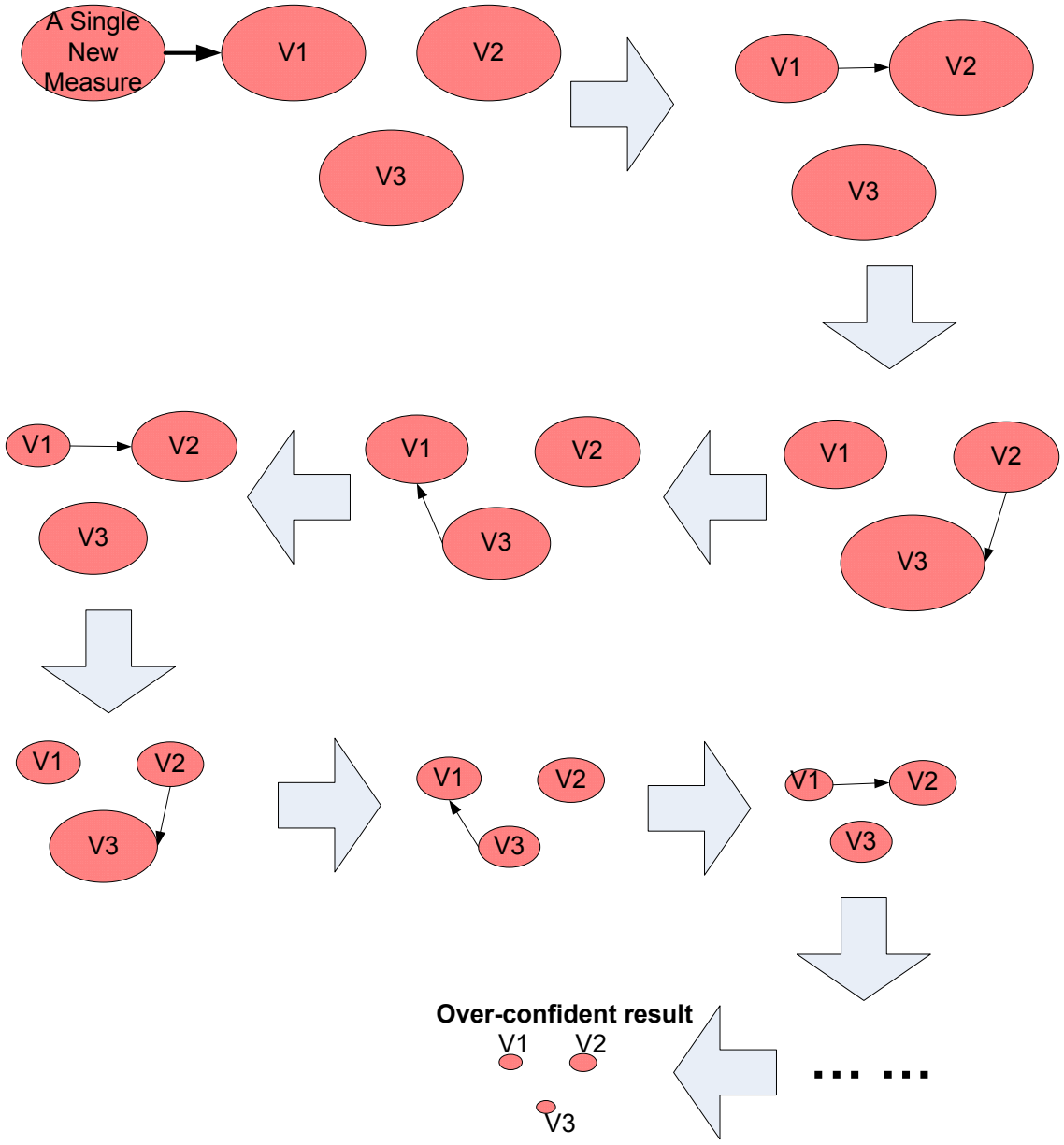


Figure 2.8 Circular reasoning and over-convergence problem

Since the over-convergence problem is due to inter-estimates correlation, a natural idea for avoiding this problem is to control inter-estimates correlation, which is realized by **monitoring and controlling the data flow within vehicle networks**. This practice is popularly adopted for dealing with inter-estimates correlation in cooperative localization; here are some examples.

Simple heuristic rules: The authors in [Howard *et al.* 2003] propose a heuristic method based on a dependency tree which establishes for each distribution only one parent distribution and zero or more child distributions. A distribution can not be used to update its ancestors, but may be used to update its descendants. The relationship among these distributions may change according to the arrival of new observations. The authors in [Fox *et al.* 2000] propose some heuristic rules which function similarly with the idea of the dependency tree. The limitation of these heuristic methods is that they do not have complete monitoring or controlling of data flow; the risk of circular reasoning and over-convergence may still exist.

Complicated data transfer scheme: The authors in [Leung *et al.* 2009] propose a data transfer scheme which enables distributed robots to obtain delayed estimates that are comparable with centralized estimates; the delay length depends on the evolution of the communication graph over time. One undesirable aspect of this method is the uncertainty of the availability of the fused estimates, which might not satisfy real-time requirements; besides, the communication requirement of this method is demanding, due to the large pedigrees of data that have to be relayed within the networks.

Independent estimates exchange: The authors in [Karam *et al.* 2006a] [Karam *et al.* 2006b] propose state exchange based method which only allows independent estimate (estimate maintained by each vehicle using its own sensor measurements) to be shared within vehicle networks; thus the risk of over-convergence can be thoroughly removed. The method can achieve locally centralized performance. On the other hand, a major disadvantage for this method is: a vehicle can not benefit from other vehicles that are outside its directly visible neighborhood. Besides, one more set of estimate should be maintained by each vehicle.

We can still strive in the direction of monitoring and controlling the data flow within vehicle networks, with a goal of finding certain data transfer scheme that can keep desirable balance among its performance, efficiency, complexity, and feasibility.
On the other hand, we could reflect a question: do we have to monitor and control the data flow for cooperative localization purpose? In above methods, the reason why we have to monitor and control the data flow is that the fusion methods used in above methods, such as Kalman filter, can not guarantee yielding consistent estimate when fusing correlated data (simply speaking, *consistent* means that the estimate should not

be over-confident compared with the ground-truth; detailed explanation is postponed to CHAPTER 3).

What if there exist fusion methods that yield consistent estimate even facing unknown degree of inter-estimates correlation? The covariance intersection filters [Julier & Uhlmann 1997] [Julier & Uhlmann 2001] are such kind of fusion method.

Recently, we propose in [Li & Nashashibi 2012a] **a general architecture of cooperative localization using split covariance intersection filter** [Julier & Uhlmann 2001]. The basic idea is: each vehicle maintains an estimate of a decomposed group state and this estimate is shared with neighboring vehicles; the estimate of the decomposed group state is updated with both the sensor data of the ego-vehicle and the estimates sent from other vehicles, based on the split covariance intersection filter. Detailed description of this method will be presented in CHAPTER 3.

This proposed method has several merits for implementing cooperative localization: First, the vehicles are exempted from complicated techniques of monitoring and controlling data flow within vehicle networks and the programming architecture for cooperative localization becomes simpler. Second, the risk of over-convergence can be essentially removed, because the risk is removed directly by the estimation method itself. Third, communication requirement are comparatively low, because no pedigree of information is needed to be explicitly relayed within vehicle networks.

2.4.3 Vehicle-to-Vehicle Relative Pose Estimation

An essential requirement for multiple vehicles to realize cooperative localization is their ability to estimate vehicle-to-vehicle (V2V) relative pose among them. In existing cooperative localization methods, V2V relative pose estimation is usually performed directly, i.e. a vehicle directly detects another vehicle and computes their relative pose from the detection result. The realization of a direct V2V relative pose estimation method requires dealing with three sub-problems: 1) *detection*: the vehicle should detect and recognize the existence of other vehicles from perceptive data (such as range data and vision data); 2) *data association*: a detection result should be correctly associated with a vehicle in cooperation; 3) *relative pose computation*: compute the relative pose from the detection result. In indoor environment applications where the number of cooperative vehicles is limited and the scenario is comparatively static, certain *ad hoc* patterns (with special colors, shapes etc) can be designed and installed on vehicles to facilitate dealing with these sub-problems [Fox *et al.* 2000] [Howard *et al.* 2003] [Howard *et al.* 2006].

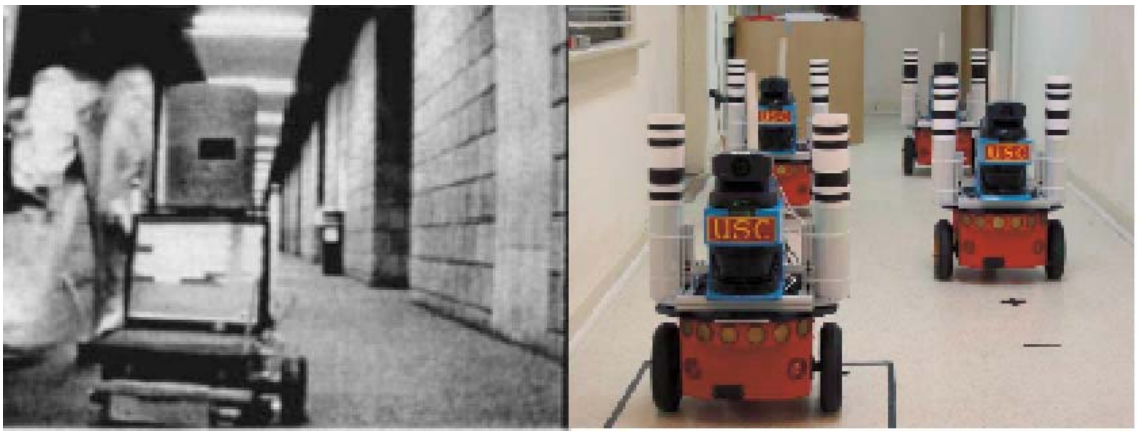


Figure 2.9 Intelligent vehicles (robots) with special patterns (Left picture from [Fox *et al.* 2000]; right picture from [Howard *et al.* 2006])

On the other hand, these sub-problems faced by direct V2V relative pose estimation methods are not easy to tackle in outdoor traffic environment which is dynamic, unpredictable, and full of partial or complete occlusions caused by dynamic entities. Reliable vehicle detection in outdoor environment is still a challenging problem which deserves continuous research works. Data association is an even challenging problem, especially when the vehicles have low-accuracy self-localization ability (with a normal GPS, the vehicle localization error can be a dozen of meters). Special patterns might be used to facilitate vehicle detection and data association, as in indoor environment. However, there are thousands of vehicle systems in traffic environment. The task of designing proper patterns to distinguish such huge number of vehicle systems is not trivial; besides, occlusions might cause miss detection and false detection of these patterns. Even if vehicle detection and data association are performed correctly, it is still difficult to extract accurate geometric information of the detected vehicle, because the detection result always corresponds to partial contour (sometimes even irregular) of the detected vehicle.

Recently, we propose in [Li & Nashashibi 2012b] an **indirect vehicle-to-vehicle (V2V) relative pose estimation method**. The basic idea is as follows: each vehicle maintains in real-time a dynamic local map [Wang 2004] [Vu 2009] whose spatial relationship with respect to the vehicle can be rather precisely computed. When two vehicles are in cooperation, their local maps are spatially associated via maps merging. As the spatial relationship between each vehicle and its local map has already been computed, the relative pose between the two vehicles can be indirectly computed by compounding a chain of transformations.

In the indirect V2V relative pose estimation method, the challenging problems of vehicle detection, data association, and relative pose computation in direct V2V relative pose estimation methods are implicitly solved during local maps merging. For local maps of a scale such as 80 meters, large and stable objects (buildings, infrastructures etc) are usually the dominating factors, which contribute to successful local maps merging.

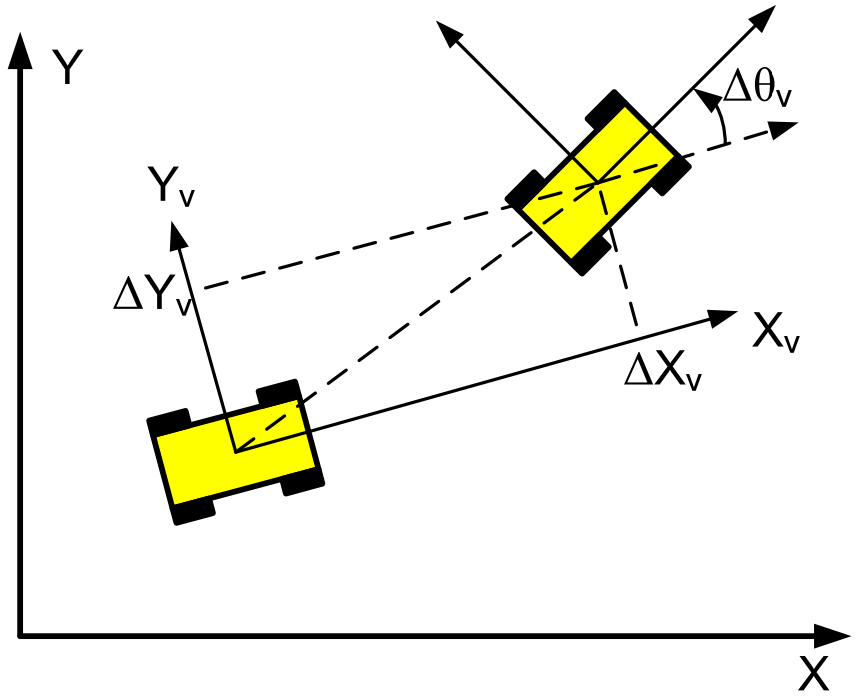


Figure 2.10 The relative pose of one vehicle with respect to another vehicle

2.5 Cooperative Local Mapping and Moving Objects Detection

2.5.1 Perception Representation

By so far, we have been often using the term *perception* whose meaning can be rather general. As we briefly specified in Introduction chapter, we deal with issues of vehicle localization and local environment mapping; in other words, we mainly deal with the perception concerning spatial properties of the vehicles and the environment.

Therefore, in this dissertation, we could fairly bear in mind that the *perception* concretely indicates certain localization result and local mapping result including moving objects. If we discuss some issues in a general manner, we would continue using the general term *perception*; if we discuss some concrete practice, we would use the specific terms *vehicle pose* and *local map* etc.
The representation for vehicle pose is rather fixed; normally, vehicle pose is represented by a triplet, two elements for **vehicle position** and one element for **vehicle orientation** (heading direction). In contrast, there is considerable flexibility for environment representation. Here are several typical methods for environment representation (we usually only care about two-dimensional environment representation, as the vehicles operate only in horizontal plane).

Direct representation: the environment can be represented by a registered list of raw perception scans (or frames), for example, a registered list of raw laser scans [Gutmann & Schlegel 1996] [Lu & Milios 1997b]. Direct representation exempts the perception system from making conversion on raw perception measurements. On the other hand, direct representation has several disadvantages: the perception uncertainties are usually not modeled. Besides, direct representation only models the environment part that the sensor can directly measure; for example, free space can not be modeled by direct representation. Moreover, management of direct representation is comparatively complicated; for example, how to deal with overlapping part of raw scans that contains redundant information (considering that different scans have certain degree of perception inconsistency about the overlapping perception part due to perception errors)? How to organize the data (structure) in direct representation so that we can conveniently and efficiently access and use the data?

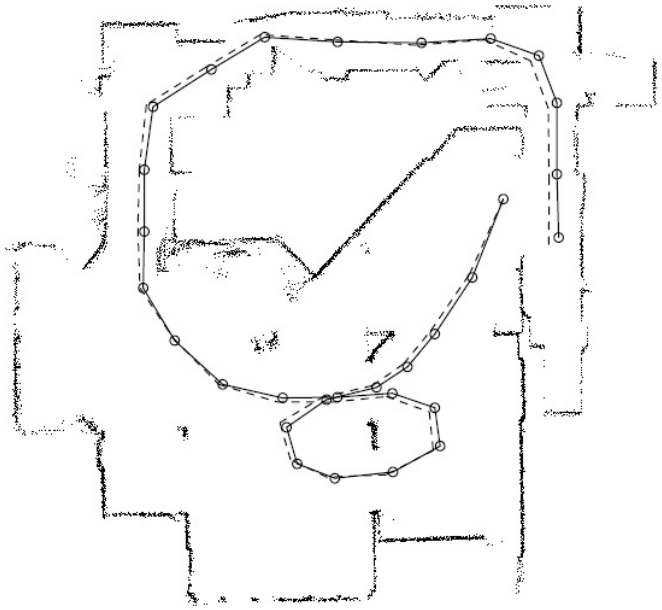


Figure 2.11 Direct representation (from [Lu & Milios 1997b])

Features based representation: The environment can also be represented by a sparse map of features that are extracted from raw perception data. The features can be artificial landmarks [Montemerlo *et al.* 2002]; they can be natural landmarks such as trees outdoor [Guivant *et al.* 2000] or line segments indoor [Cox 1991]; they can also be of abstract form, picked out by certain specific detection algorithms [Royer 2006] [Gil *et al.* 2010]. An advantage of features based representation lies in its low memory requirement and its flexibility to adjust its elements.

However, an apparent disadvantage for features based representation is that it lacks the ability to represent general unstructured environment, because it requires extraction of certain environment features and is only able to represent these features. Therefore, its application is usually limited to rather static environment where certain features are extractable.

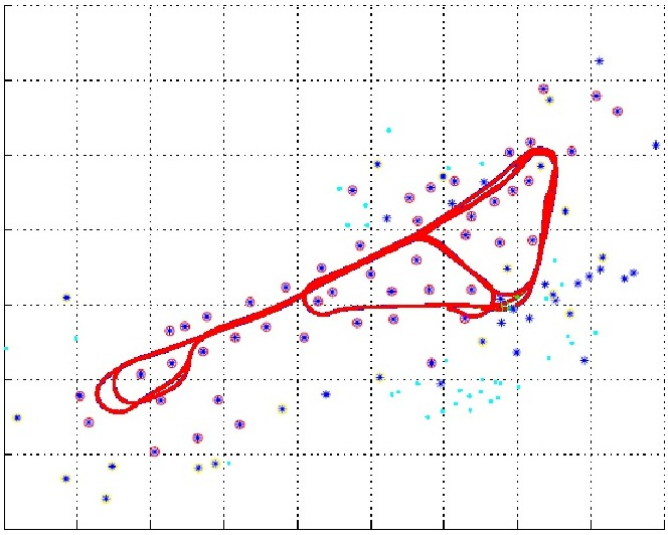


Figure 2.12 Features based representation (from [Guivant *et al.* 2000])

Occupancy grid based representation: the occupancy grid [Elfes 1989] is a two-dimensional lattice which divides the environment space into rectangular cells; each cell is associated with a real value in the unit interval [0, 1], where the cell value represents the degree of the cell being occupied by or free of object. The cell value 0.5 represents the cell being in unknown state, neither occupied nor free. For cell value larger than 0.5, the larger the cell value is, the more likely the cell is occupied. For cell value smaller than 0.5, the smaller the cell value is, the more likely the cell is free.

The occupancy grid based representation has several merits: most importantly, it has the ability to represent general unstructured environment. Here, the *general* implies not only the place with objects but also the place free of objects. Besides, the occupancy grid based representation is similar to our daily-life maps, which makes it more suitable to our normal thinking habit for dealing with maps. It can also be directly and conveniently visualized.

The occupancy grid based representation is a sort of dense representation for the environment. If we aim at establishing a global map for a large environment area, the memory requirement increases quickly, which might make the occupancy grid based representation intractable. On the other hand, global mapping is not necessary for many real-time vehicle operations; as the environment is always changing, even if the vehicle stores the map of a passed place, this map would be of little value for the vehicle when it re-passes the same place in future. Instead, it is better for a vehicle to maintain in real-time a local map that the vehicle has to deal with directly and immediately. For local

mapping, the occupancy grid based representation can easily be controlled at tractable size.

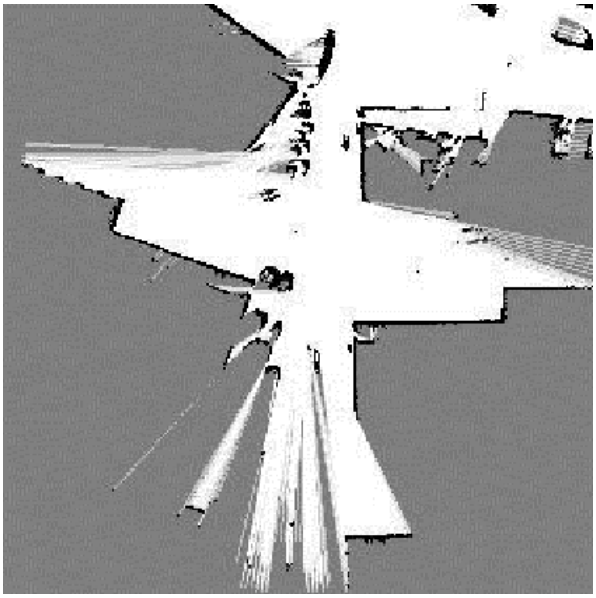


Figure 2.13 Occupancy grid based representation

2.5.2 Perceptions Association

The practice of cooperative perception is NOT simply the process of data sharing through inter-vehicle communication. For a vehicle, besides receiving the perceptions shared by other vehicles, it also has to be able to utilize the shared perceptions. For this purpose, a most fundamental problem is the problem of **perceptions association** i.e. **given two perceptions from different vehicles, how to establish consistent (spatial) relationship between the two perceptions**?

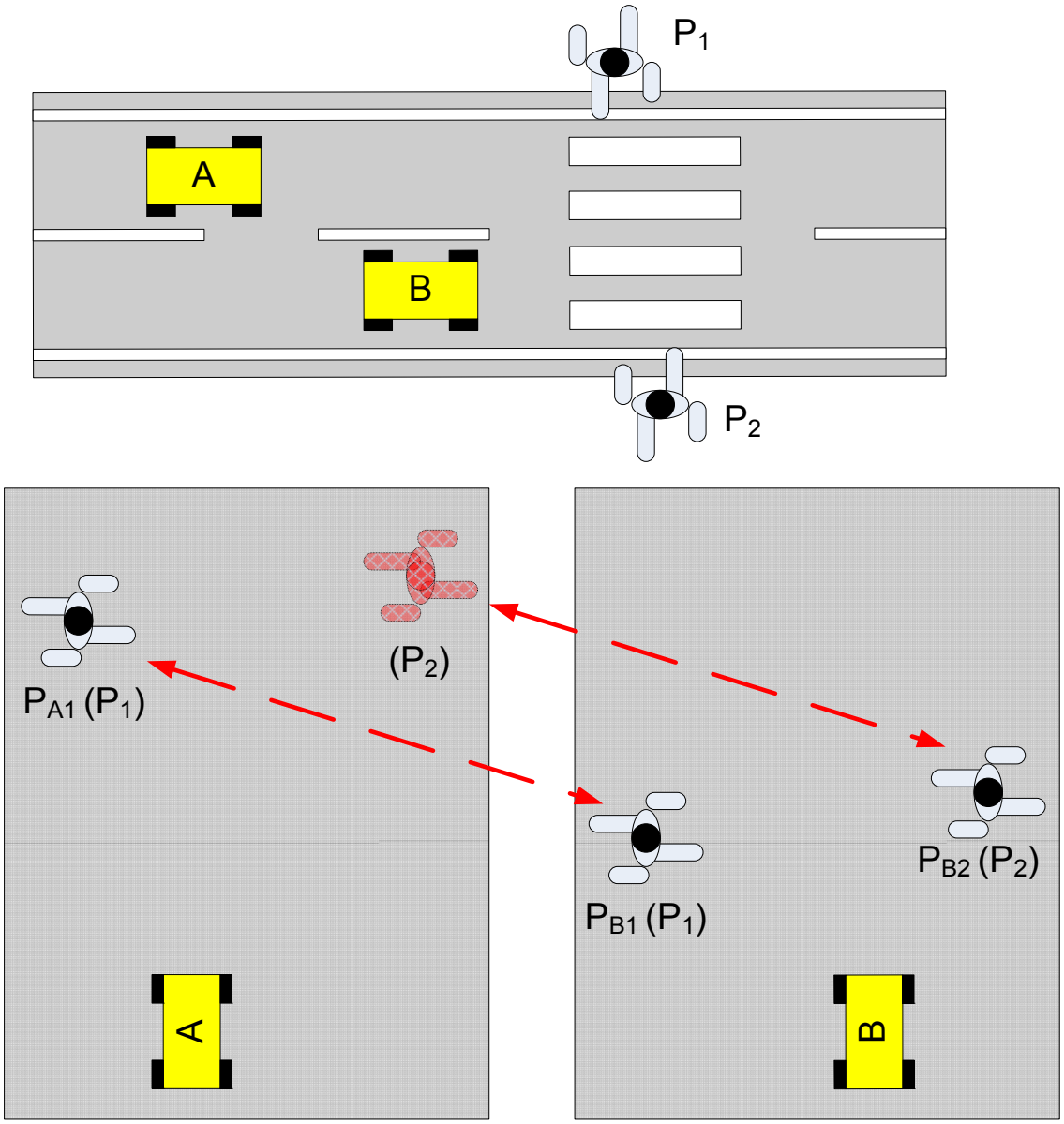


Figure 2.14 Perceptions association: the object you see corresponds to what (or where) I see?

For example, given two vehicles (A and B) and two pedestrians (P_1 and P_2), as illustrated in Figure 2.14. For vehicle A, pedestrian P_2 is occluded by vehicle B, so it can only perceive pedestrian P_1, as represented by P_{A1} in the bottom-left sub-figure. Vehicle B can perceive both pedestrian P_1 and P_2, as represented by P_{B1} and P_{B2} in the bottom-right sub-figure. Suppose vehicle B share its perception (including P_{B1} and P_{B2}) with vehicle A; how does vehicle A establish between its own perception and the perception of vehicle B a consistent relationship which can spatially associate P_{A1} and

P_{B1} correctly? This belongs to what perceptions association handles. Without correct perceptions association, vehicle A can not have any meaningful inference from the data shared by vehicle B.

On the whole, there are two categories of methods for performing perceptions association: vehicle pose estimation based methods, and perceptions consistency based methods. These two categories of methods will be discussed in following sub-sections.

Since perceptions are a kind of data, the scope of *perceptions association* belongs to a more general scope of *data association*. However, *data* do not necessarily mean perceptions; for example, the short-term motion intentions of the vehicles, their long-term tasks planning, and the knowledge of specific traffic rules in certain particular area, they all are kinds of data. Therefore, the expression of *perceptions association* is preferred here in order to highlight and specify our concern on associating different perceptions (especially of different vehicles).

2.5.3 Vehicle Pose Estimation based Methods

If each vehicle can be precisely localized in a common global reference, then perceptions association is only a trivial issue of transforming the local maps of different vehicles into a common reference based on the precise vehicle localization results. In reality, however, we can not expect every vehicle possessing precise localization ability. For a vehicle equipped with low-cost GPS, the global localization error can be as large as ten meters in the position component. Even for a vehicle with high accuracy RTK-GPS, it might encounter occasions of signal degradation. Therefore, perceptions association based on vehicles global localization results is generally impractical in real implementation.

One possible practice for perceptions association is based on direct vehicle-to-vehicle relative pose estimation [Madhavan *et al.* 2004] [Howard 2006] [Carlone *et al.* 2011] [Nerurkar *et al.* 2009]. If a vehicle can determine the relative pose of another vehicle, then it can spatially relate the perception of this other vehicle to its own perception. On the other hand, as discussed in Section 2.4.3, the realization of a direct vehicle-to-vehicle relative pose estimation method requires dealing with the sub-problems of *detection*, *data association*, and *relative pose computation*, which are NOT easy to handle in outdoor traffic environment.

2.5.4 Perceptions Consistency based Methods

Perceptions consistency based methods do not require an accurate estimate of vehicle pose or vehicle-to-vehicle relative pose; they try to associate different perceptions directly, according to certain consistency measure between the perceptions themselves.

The idea of perceptions consistency based perceptions association has been intensively exploited in literature, from different perspectives according to concrete applications. If we treat the perception of a vehicle generally as one scan of data, then the process of perceptions consistency based perceptions association becomes the process of **scan matching** [Lu & Milios 1997a], which tries to find correct alignment of two scans by maximizing certain consistency measure. If the perception concerned is a map, then the process of perceptions consistency based perceptions association becomes the process of **maps merging** [Birk & Carpin 2006], which tries to find correct alignment of two maps also by maximizing certain consistency measure.

Various methods have been proposed in literature. We could categorize these methods according to different criteria; for example, concerning environment representation, some methods deal with direct representation [Lu & Milios 1997a], some methods deal with features based representation [Cox 1991], and some methods deal with occupancy grid based representation [Birk & Carpin 2006]. If we examine the procedure form of these methods, we could roughly assign them into three main categories: **features based methods**, **iterative closest point (ICP) methods**, and **overall-direct optimization based methods**.

Features based method: For applications in structured indoor environment, one can fairly assume the existence of certain natural features or artificially deposited features; these features can be extracted first and then the association is carried out on these features [Cox 1990] [Cox 1991] [Grossmann & Poli 2001] [Dedeoglu & Sukhatme 2000]. The availability of features would facilitate the process of perceptions association. However, similar to features based map representation, features based methods are usually limited to the environment where certain features (concrete or abstract) are extractable.

It is worthy noting that features based methods are not necessarily juxtaposed with features based map representation. For example, occupancy grid based representation can be used, where features are extracted from the occupancy grid map only for association purpose [Saeedi *et al.* 2011] [Topal *et al.* 2010].

Iterative closest point (ICP) method: the ICP method is a popular architecture to associate perceptions of general representation, without extraction and use of special features. It was originally introduced in [Besl & McKay 1992] (similar idea was introduced in [Chen & Medioni 1992]); a survey for the variants of the original ICP method can be referred to [Rusinkiewicz & Levoy 2001]. The ICP method consists in an iteration of two steps: first, a new set of correspondences between the two perceptions is established tentatively based on the closest point rule; second, the estimate of the perceptions alignment is updated by minimizing the overall distances between the points in the new set of correspondences. Given a suitable initial value, the estimate of the perceptions alignment will normally converge after a number of iterations of the two steps.

The core of the ICP method is the practice of establishing point-to-point correspondences based on the closest point rule; originally, point-to-point distance is computed using the Euclidean distance measure. Later, more general distance measures [Lu & Milios 1997a] [Minguez *et al.* 2006] have been incorporated in ICP to establish correspondences, which might better capture the true alignment of perceptions.

Overall-Direct optimization based method: this kind of method normally consists of two parts: First, an objective function in terms of the perceptions alignment variables is defined as the measure to characterize the overall consistency degree between the two perceptions; second, the defined objective function is optimized using certain optimization technique. For example, [Biber & Strasser 2003] propose a Normal Distribution Transform to transform raw scan points into a collection of normal distributions which serves as the consistency measure; then the Newton's algorithm is used as the optimization technique. [Birk & Carpin 2006] propose a method for occupancy grid maps merging; in this method, the consistency degree between two occupancy grid maps is measured by an objective function consisting of a *similarity* term and a *lock term*; then random walk algorithm is used to search the optimum.

Despite that features based methods and ICP methods also reflect the spirit of optimization in terms of perceptions consistency, they are essentially different from overall-direct optimization based methods in how they put this spirit into practice. In features based methods, each feature normally characterizes certain local property of the environment; when we associate the features, we actually deal with local consistency or partial consistency between the perceptions. In contrast, overall-direct optimization based methods deal with overall consistency between the perceptions. ICP methods also deal with overall consistency between the perceptions, yet their measure on perceptions

consistency changes during the association process; when a new measure is established, they have to re-execute an optimization process. In contrast, in overall-direct optimization based methods, once the measure on perceptions consistency is established, only one optimization process is needed; in other words, the optimization architecture is more direct. Naturally, above explanations also explain the meaning of *overall* and *direct* in overall-direct optimization based methods.

We could also examine the *problem size* handled by the perceptions consistency based methods. There are three levels of problem size: **problem at small**, **problem at middle**, and **problem at large**, which are roughly classified according to the uncertainty level of initial estimate.

Problem at small: initial estimate of the perceptions association is available and the initial estimate is close enough to the correct association.

Problem at middle: initial estimate of the perceptions association is available, yet the initial estimate is not close to the correct association. According to the initial estimate, we can know there is a correct association between the perceptions; what to do is to find the correct association.

Problem at large: No initial estimate of the perceptions association is available at all. We even do not know whether there is any relationship between the perceptions. For example, suppose two vehicles operate in separated areas and share their perceptions, then their perceptions have no relationship at all, not to mention an association between them.

Above categorization is not strict; it is difficult to give concrete criteria to specify these three levels of problem size. We could conveniently think them via an analogy between perceptions association and a general optimization problem. For the optimization problem, if the starting point is already in the local unimodal neighborhood of the global optimum, then it is problem at small. If the starting point is not close to the global optimum (not in the unimodal neighborhood of the global optimum) but we know there is a global optimum in certain range with respect to the starting point, then it is problem at middle. If we have no idea about a possible range for the global optimum and even do not know whether there is a global solution, then it is problem at large.

ICP methods are usually limited to problem at small, because they need initial estimate that is good enough; otherwise, they would easily suffer from the problem of converging to a local optimum that is wrong solution. In contrast, features based methods and overall-direct optimization based methods have the potential to deal with all levels of problem size.

2.5.5 Local Occupancy Grid Maps Merging

For our research context, we prefer the occupancy grid based environment representation for its ability to represent general unstructured outdoor environment. Besides, as we deal with real-time vehicle perception, each vehicle only needs to maintain a local occupancy grid map. Then for two vehicles to perform cooperative mapping, we have to solve the fundamental problem of associating their local occupancy grid maps or local occupancy grid maps merging.

Here, the problem of local occupancy grid maps merging belongs to problem at middle. On one hand, initial estimate of the association can be provided by GPS based global vehicle localization results. Despite of GPS positioning error which might be ten meters, we can at least know whether the vehicles are in the common area and whether their perceptions are associable. Therefore, the problem does not belong to problem at large. On the other hand, as GPS positioning error can be as large as ten meters, the initial estimate might not be close enough to the correct association; local optimization techniques tend to fail and get stuck into a wrong local optimum. Therefore, the problem does not belong to problem at small either.

For occupancy grid maps merging, we can not rely on the ICP methods, as they are limited to problem at small; we do not want to rely either on extraction of certain map features. Instead, we prefer to follow the overall-direct optimization based methods. As mentioned in previous sub-sections, [Birk & Carpin 2006] propose a framework for merging different occupancy grid maps directly via optimization on an objective function. In [Birk & Carpin 2006], the objective function F_c consists of a *similarity* term and a *lock* term: the similarity term which is based on a distance-map represents the overall distances between the maps to-be-merged; the lock term is a part heuristically added to counteract the over-fitting effect.

This objective function in [Birk & Carpin 2006] has two major disadvantages: first, the parameter c_{lock} in the heuristically added lock term has to be tuned empirically according to concrete scenarios. Second, this objective function is susceptible to maps inherent inconsistency i.e. maps inconsistency that still exists even if the maps to-be-merged are aligned correctly. Maps inherent inconsistency can be caused by dynamic

entities which are common in outdoor environment. Maps inherent inconsistency can also be caused by the inconsistency of perception poses at different vehicles; for example, the same environment might appear noticeably different if it is scanned by laser scanners at different heights. For the objective function in [Birk & Carpin 2006], maps inherent inconsistency would cause drastic value change in the distance-map based similarity term and false counting of *agreement* and *disagreement* in the lock term.

Recently, we propose in [Li & Nashashibi 2012c] a new method for occupancy grid maps merging: an objective function based on occupancy likelihood is introduced to measure the consistency degree of maps alignment; genetic algorithm implemented in a dynamic scheme is adopted to optimize the objective function. The proposed objective function only takes into account the consistent part of the maps to-be-merged; thus it is insensitive to maps inherent inconsistency. For local maps of enough size, stable and consistent objects (buildings, infrastructures etc) are usually the dominating factors, which always contribute to successful local maps merging.

2.5.6 Moving Objects Detection

One motivation for performing cooperative perception is to better detect moving objects, such as vulnerable road users, as illustrated in Figure 1.7. We care about moving objects because it is them who are likely to be involved in a traffic accident.

Moving objects detection (and tracking) has been intensively and extensively researched since several decades ago. However, we do not intend to discuss much about this subject. As we have stated previously, we would rather focus on the general mechanisms which enable the realization of cooperative perception functionalities. Perceptions association based on the local occupancy grid maps merging is such kind of general mechanism, through which the moving objects detection result of an intelligent vehicle can be shared to and used by another intelligent vehicle, no matter what moving objects detection method is concretely adopted.

To demonstrate a complete application of cooperative detection on moving objects, we incorporate two basic moving objects detection methods, namely *consistency-based detection* and *moving object map based detection* [Wang 2004] [Vu 2009]. More sophisticated moving objects detection methods can also be incorporated into the local occupancy grid maps merging based scheme of cooperative local mapping and moving objects detection.

2.6 Cooperative Augmented Reality

2.6.1 Augmented Reality Effect of 'Seeing' Through Front Vehicle

We propose in [Li & Nashashibi 2011a] a brand new idea of *cooperative augmented reality* which utilizes cooperative perception results to realize a special augmented effect. More specifically, in [Li & Nashashibi 2011a], we realize the effect of 'seeing' through front vehicle, as illustrated in Figure 2.15. To realize this, first, the 2D range data of the two vehicles is associated using the iterative closest point method. Second, 3D perspective transform between the visual perceptions of the two vehicles is performed, based on approximate estimation of the visual perception depth with the help of 2D range perception.

Figure 2.15 Augmented reality: 'see' through the front vehicle

In [Li & Nashashibi 2011a], we used the iterative closest point method for perceptions association, as the GPS used provided rather accurate positioning result which can further provide good initial estimate of the association. We improve the works in [Li &

Nashashibi 2011a] by incorporating the proposed local occupancy grid maps merging method for perceptions association, which can work even facing large GPS position errors.

2.7 Summary

We have introduced several sorts of sensors, namely GPS, laser scanner, camera, and motion sensor, which have been commonly used for single intelligent vehicle operation; with these sensors, an intelligent vehicle can possess fairly complete perception ability towards itself and the environment. We have reviewed the Bayesian filter framework that has been commonly used for recursive state estimation; we have also reviewed several recursive estimation methods that are derived from the Bayesian filter framework based on different kinds of approximations. We have discussed in details the fundamental problems and the state-of-the-art methods concerned in cooperative localization, and cooperative local mapping and moving objects detection. Based on these discussions, we propose a general architecture of cooperative localization using split covariance intersection filter, an indirect vehicle-to-vehicle relative pose estimation method, and a new method for occupancy grid maps merging to handle the fundamental problems in cooperative localization, and cooperative local mapping and moving objects detection. We propose a brand new idea of cooperative augmented reality which utilizes cooperative perception results to realize certain augmented effect. The proposed methods (some of them have been published) will be detailed in the following chapters.

CHAPTER 3 Cooperative Localization

Résumé

Nous avons fourni une solution pour la localisation coopérative multi-véhicules. Nous avons rappelé le concept de consistance de l'estimation ainsi que le SCIF. Nous avons présenté plusieurs formes de ce filtre avec leurs dérivations et une preuve originale pour la consistance de la fusion de ce filtre. Nous avons introduit plusieurs fonctionnalités de base comme la condition pour réaliser la localisation coopérative. Nous avons introduit une architecture générale de localisation coopérative en utilisant un SCIF. Puisque l'architecture est décentralisée, nous avons décliné l'approche dans le cadre de la localisation d'un véhicule intelligent en s'appuyant sur ses capteurs de mouvement. Nous explicitons ainsi la manière dont il peut mettre à jour son estimation d'état en utilisant ses propres mesures de positionnement absolu, ainsi que la mise à jour de son estimation d'état avec les données partagées avec les véhicules voisins. Nous avons présenté la stratégie d'estimation indirecte de du positionnement relatif Véhicule-à-Véhicule.

3.1 Introduction

In this chapter, we describe in details our solution of multi-vehicles cooperative localization. First, we review the split covariance intersection filter that yields consistent fusion estimates even facing unknown degree of inter-estimates correlation. Then we make an abstraction of the basic functionalities that are commonly available for intelligent vehicle systems; based on these abstracted functionalities, we introduce a general architecture of cooperative localization using split covariance intersection filter. Afterwards, we present the indirect vehicle-to-vehicle relative pose estimation strategy which enables a feasible realization of cooperative localization in reality.

3.2 Consistent Fusion: Split Covariance Intersection Filter

3.2.1 Estimate Consistency

Given an estimate $\{\mathbf{X}, \mathbf{P}\}$ where $\mathbf{X}$ denotes the estimated state vector and $\mathbf{P}$ denotes the estimated covariance matrix. Let $\mathbf{P}^*$ denote the covariance of the true errors of the estimate $\mathbf{X}$, i.e.

$$\mathbf{P}^* = E[\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T] = E[(\mathbf{X} - \overline{\mathbf{X}})(\mathbf{X} - \overline{\mathbf{X}})^T]$$

Then **consistency** characterizes a property of an estimate that the estimated covariance matrix is no smaller than the true covariance of the estimated state vector; in mathematical terms:

$$\mathbf{P} - \mathbf{P}^* \geq \mathbf{0}$$

In simple words, an estimate is consistent if it is NOT over-confident.

3.2.2 Inconsistent Fusion of the Kalman Filter

Given two source estimates $\{\mathbf{X}_i, \mathbf{P}_i\}(i=1,2)$ to-be-fused; $\mathbf{X}_i$ denotes the estimated state vector and $\mathbf{P}_i$ denotes the estimated covariance matrix. Both estimates are assumed consistent:

$$\mathbf{P}_1 - \mathbf{P}_1^* \geq \mathbf{0}$$

$$\mathbf{P}_2 - \mathbf{P}_2^{*} \geq \mathbf{0}$$

Let their fusion estimate be denoted as $\{\mathbf{X},\mathbf{P}\}$. Normally, we hope that the fusion estimate can also be consistent; we do not want to establish any extra confidence on the fusion estimate than what the source estimates can convey.
Consider the Kalman filter which is popular in many applications; the basic formula of the Kalman filter can be written as:

$$\begin{aligned}\mathbf{P}^{-1} &= \mathbf{P}_1^{-1} + \mathbf{P}_2^{-1} \\ \mathbf{X} &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{P}_2^{-1}\mathbf{X}_2)\end{aligned}$$

The effectiveness of the Kalman filter is based on the assumption that the two source estimates are independent of each other. However, if there is correlation between them, the Kalman Filter might yield inconsistent fusion estimate. For example, let $\{\mathbf{X}_1,\mathbf{P}_1\}$ and $\{\mathbf{X}_2,\mathbf{P}_2\}$ are exactly two copies of the same estimate; naturally, a reasonable fusion of them will yield an estimate the same to them, especially satisfying $\mathbf{P}=\mathbf{P}_1=\mathbf{P}_2$. In contrast, when they are fused with the Kalman filter, the fused covariance $\mathbf{P}$ will become a half of $\mathbf{P}_1$ (or $\mathbf{P}_2$), which is obviously inconsistent.

3.2.3 Split Covariance Intersection Filter (Split CIF)

The authors in [Julier & Uhlmann 1997] propose a new data fusion method named *covariance intersection*, which takes a convex combination of the means and covariances of the source estimates in the information space. The covariance intersection filter is theoretically guaranteed to yield consistent results. The formula of the covariance intersection filter can be written as:

$$\begin{aligned}\mathbf{P}^{-1} &= (\mathbf{P}_1 / w)^{-1} + (\mathbf{P}_2 /(1-w))^{-1} \\ \mathbf{X} &= \mathbf{P}[(\mathbf{P}_1 / w)^{-1}\mathbf{X}_1 + (\mathbf{P}_2 /(1-w))^{-1}\mathbf{X}_2)\end{aligned}$$

This original covariance intersection filter (CIF) has a drawback of yielding pessimistic estimate, because it treats the source estimates as being totally correlated and neglects possible independent information in them. In [Julier & Uhlmann 2001] the generalized form of the covariance intersection filter, i.e. the *split covariance intersection filter*

(Split CIF) is introduced, which provides the ability to incorporate and maintain known independent information in the estimates.

Given two consistent source estimates: $\{\mathbf{X}_1, \mathbf{P}_{1d}+\mathbf{P}_{1i}\}$ and $\{\mathbf{X}_2, \mathbf{P}_{2d}+\mathbf{P}_{2i}\}$, where the covariance components $\mathbf{P}_{1d}$ and $\mathbf{P}_{2d}$ represent the maximum degree to which these estimates are possibly correlated with each other or others; the covariance components $\mathbf{P}_{1i}$ and $\mathbf{P}_{2i}$ represent the known degree of their absolute independence. Let the fusion estimate be denoted as $\{\mathbf{X}, \mathbf{P}_d+\mathbf{P}_i\}$, also with its covariance separated as two parts: $\mathbf{P}_d$ and $\mathbf{P}_i$ respectively represent the correlated and the independent part. The formula of the split covariance intersection filter can be written as:

$$\begin{aligned}
\mathbf{P}_1 &= \mathbf{P}_{1d} / w + \mathbf{P}_{1i} \\
\mathbf{P}_2 &= \mathbf{P}_{2d} /(1-w) + \mathbf{P}_{2i} \\
\mathbf{P}^{-1} &= \mathbf{P}_1^{-1} + \mathbf{P}_2^{-1} \\
\mathbf{X} &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{P}_2^{-1}\mathbf{X}_2) \\
\mathbf{P}_i &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{P}_2^{-1}\mathbf{P}_{2i}\mathbf{P}_2^{-1})\mathbf{P} \\
\mathbf{P}_d &= \mathbf{P} - \mathbf{P}_i
\end{aligned} \tag{3-1}$$

Where w belongs to the interval [0, 1] and any choice of w in this interval guarantees the consistency of the fusion estimate (see the proof below). In practice, w can be determined by optimizing an objective function in terms of w such as the determinant of the new covariance [Julier & Uhlmann 2001].

Proof: Let $\mathbf{X}_1=\mathbf{X}_{1d}+\mathbf{X}_{1i}$ and $\mathbf{X}_2=\mathbf{X}_{2d}+\mathbf{X}_{2i}$, where $\mathbf{X}_{1d}$ and $\mathbf{X}_{2d}$ correspond to the correlated components; $\mathbf{X}_{1i}$ and $\mathbf{X}_{2i}$ correspond to the independent components. For each source estimate, **its correlated component and independent component are both consistent**, i.e.

$$\mathbf{P}_{1d} \geq E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}^{T}]; \mathbf{P}_{1i} \geq E[\tilde{\mathbf{X}}_{1i}\tilde{\mathbf{X}}_{1i}^{T}]$$
$$\mathbf{P}_{2d} \geq E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}^{T}]; \mathbf{P}_{2i} \geq E[\tilde{\mathbf{X}}_{2i}\tilde{\mathbf{X}}_{2i}^{T}]$$

We examine the independent component of the fusion estimate $\mathbf{X}$:

$$
\begin{aligned}
&\mathbf{P}_i - E[\tilde{\mathbf{X}}_i \tilde{\mathbf{X}}_i{}^T] \\
&= \mathbf{P}(\mathbf{P}_1{}^{-1}\mathbf{P}_{1i}\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}\mathbf{P}_{2i}\mathbf{P}_2{}^{-1})\mathbf{P} - E[\tilde{\mathbf{X}}_i \tilde{\mathbf{X}}_i{}^T] \\
&= \mathbf{P}(\mathbf{P}_1{}^{-1}\mathbf{P}_{1i}\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}\mathbf{P}_{2i}\mathbf{P}_2{}^{-1})\mathbf{P} \\
&\quad - \mathbf{P}\{\mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1i}\tilde{\mathbf{X}}_{1i}{}^T]\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2i}\tilde{\mathbf{X}}_{2i}{}^T]\mathbf{P}_2{}^{-1}\}\mathbf{P} \\
&= \mathbf{P}\{\mathbf{P}_1{}^{-1}(\mathbf{P}_{1i} - E[\tilde{\mathbf{X}}_{1i}\tilde{\mathbf{X}}_{1i}{}^T])\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}(\mathbf{P}_{2i} - E[\tilde{\mathbf{X}}_{2i}\tilde{\mathbf{X}}_{2i}{}^T])\mathbf{P}_2{}^{-1}\}\mathbf{P} \geq \mathbf{0}
\end{aligned}
$$

We examine the correlated component of the fusion estimate $\mathbf{X}$ (for $0<w<1$):

$$
\begin{aligned}
&\mathbf{P}_d - E[\tilde{\mathbf{X}}_d \tilde{\mathbf{X}}_d{}^T] \\
&= \mathbf{P} - \mathbf{P}_i - E[\mathbf{P}(\mathbf{P}_1{}^{-1}\tilde{\mathbf{X}}_{1d} + \mathbf{P}_2{}^{-1}\tilde{\mathbf{X}}_{2d})(\tilde{\mathbf{X}}_{1d}{}^T\mathbf{P}_1{}^{-1} + \tilde{\mathbf{X}}_{2d}{}^T\mathbf{P}_2{}^{-1})\mathbf{P}] \\
&= \mathbf{P} - \mathbf{P}(\mathbf{P}_1{}^{-1}\mathbf{P}_{1i}\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}\mathbf{P}_{2i}\mathbf{P}_2{}^{-1})\mathbf{P} - \mathbf{P}\{\mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad + \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} + \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&= \mathbf{P}\{\mathbf{P}^{-1} - \mathbf{P}_1{}^{-1}\mathbf{P}_{1i}\mathbf{P}_1{}^{-1} - \mathbf{P}_2{}^{-1}\mathbf{P}_{2i}\mathbf{P}_2{}^{-1} - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&= \mathbf{P}\{\mathbf{P}_1{}^{-1}\mathbf{P}_1\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}\mathbf{P}_2\mathbf{P}_2{}^{-1} - \mathbf{P}_1{}^{-1}\mathbf{P}_{1i}\mathbf{P}_1{}^{-1} - \mathbf{P}_2{}^{-1}\mathbf{P}_{2i}\mathbf{P}_2{}^{-1} \\
&\quad - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&= \mathbf{P}\{\mathbf{P}_1{}^{-1}\frac{\mathbf{P}_{1d}}{w}\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}\frac{\mathbf{P}_{2d}}{1-w}\mathbf{P}_2{}^{-1} - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&\geq \mathbf{P}\{\mathbf{P}_1{}^{-1}(\frac{1}{w}-1)E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} + \mathbf{P}_2{}^{-1}(\frac{1}{1-w}-1)E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad - \mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} - \mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&= \frac{1}{w(1-w)}\mathbf{P}\{(1-w)^2\mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1} + w^2\mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} \\
&\quad - w(1-w)\mathbf{P}_1{}^{-1}E[\tilde{\mathbf{X}}_{1d}\tilde{\mathbf{X}}_{2d}{}^T]\mathbf{P}_2{}^{-1} - w(1-w)\mathbf{P}_2{}^{-1}E[\tilde{\mathbf{X}}_{2d}\tilde{\mathbf{X}}_{1d}{}^T]\mathbf{P}_1{}^{-1}\}\mathbf{P} \\
&= \frac{1}{w(1-w)}\mathbf{P}E\{[(1-w)\mathbf{P}_1{}^{-1}\tilde{\mathbf{X}}_{1d} - w\mathbf{P}_2{}^{-1}\tilde{\mathbf{X}}_{2d}][(1-w)\mathbf{P}_1{}^{-1}\tilde{\mathbf{X}}_{1d} - w\mathbf{P}_2{}^{-1}\tilde{\mathbf{X}}_{2d}]^T\}\mathbf{P} \geq \mathbf{0}
\end{aligned}
$$

In summary, the correlated component and the independent component of the fusion estimate are also consistent. End. □

Note: *originally in [Julier & Uhlmann 2001], no strict proof was given for the split CIF; it was not mentioned either the specification that* ***the correlated component and the independent component of an estimate are consistent respectively*** *(if so, the whole estimate is naturally consistent). From the proof that we give above, we think this specification is what consistency indeed means for split CIF*.

Notice that the split covariance intersection filter in (3-1) is compatible with the Kalman Filter: let the $\mathbf{P}_{1d}$ and $\mathbf{P}_{2d}$ be zero and (3-1) will become the same to the Kalman Filter. Therefore, the Kalman filter can be treated as a special case of the split covariance intersection filter where the two source estimates are known to be totally independent.

3.2.4 Split CIF for Partial Observation Case

Given two source estimates $\{\mathbf{X}_1, \mathbf{P}_{1d}+\mathbf{P}_{1i}\}$ and $\{\mathbf{X}_2, \mathbf{P}_{2d}+\mathbf{P}_{2i}\}$ whose meanings have been specified above, suppose the $\mathbf{X}_1$ is complete observation i.e. $\mathbf{X}_1=\mathbf{X}_{true}+noise$, whereas the $\mathbf{X}_2$ is partial observation i.e. $\mathbf{X}_2=\mathbf{H}\mathbf{X}_{true}+noise$ ($\mathbf{H}$ is not of full rank). The split covariance intersection filter can be given as in (3-2):

$$\begin{aligned}
\mathbf{P}_1 &= \mathbf{P}_{1d} / w + \mathbf{P}_{1i} \\
\mathbf{P}_2 &= \mathbf{P}_{2d} /(1-w) + \mathbf{P}_{2i} \\
\mathbf{P}^{-1} &= \mathbf{P}_1^{-1} + \mathbf{H}^T \mathbf{P}_2^{-1} \mathbf{H} \\
\mathbf{X} &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{H}^T \mathbf{P}_2^{-1}\mathbf{X}_2) \\
\mathbf{P}_i &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{H}^T \mathbf{P}_2^{-1}\mathbf{P}_{2i}\mathbf{P}_2^{-1}\mathbf{H})\mathbf{P} \\
\mathbf{P}_d &= \mathbf{P} - \mathbf{P}_i
\end{aligned} \tag{3-2}$$

Proof: Complement $\mathbf{H}$ with $\mathbf{H}_0$ to make an invertible matrix $\mathbf{H}_A$:

$$\mathbf{H}_A = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}_0 \end{bmatrix}$$

Augment $\mathbf{X}_2$ to a complete observation $\mathbf{X}_{2A}$ with the covariance estimate $\mathbf{P}_{2Ad}+\mathbf{P}_{2Ai}$, which satisfy:

$$\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}\mathbf{X}_{2A} \sim N(\begin{bmatrix}\mathbf{X}_2\\ \mathbf{0}\end{bmatrix},\begin{bmatrix}\mathbf{P}_{2d}+\mathbf{P}_{2i} & \mathbf{0}\\ \mathbf{0} & \infty\end{bmatrix})$$

Then we have:

$$\mathbf{X}_{2A} = \begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{X}_2\\ \mathbf{0}\end{bmatrix}$$

$$\mathbf{P}_{2Ad}+\mathbf{P}_{2Ai} = \begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{P}_{2d}+\mathbf{P}_{2i} & \mathbf{0}\\ \mathbf{0} & \infty\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-T}$$

$$= \underbrace{\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{P}_{2d} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-T}}_{\mathbf{P}_{2Ad}} + \underbrace{\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{P}_{2i} & \mathbf{0}\\ \mathbf{0} & \infty\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-T}}_{\mathbf{P}_{2Ai}}$$

We apply the split CIF (3-1) on $\{\mathbf{X}_1, \mathbf{P}_{1d}+\mathbf{P}_{1i}\}$ and $\{\mathbf{X}_{2A}, \mathbf{P}_{2Ad}+\mathbf{P}_{2Ai}\}$:

$$\mathbf{P}_1 = \mathbf{P}_{1d}/w + \mathbf{P}_{1i}$$

$$\mathbf{P}_{2A} = \begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{P}_2 & \mathbf{0}\\ \mathbf{0} & \infty\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-T} \quad (\mathbf{P}_2 = \mathbf{P}_{2d}/(1-w) + \mathbf{P}_{2i})$$

$$\mathbf{P}^{-1} = \mathbf{P}_1^{-1} + \mathbf{P}_{2A}^{-1}$$

$$= \mathbf{P}_1^{-1} + \begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^T\begin{bmatrix}\mathbf{P}_2^{-1} & \mathbf{0}\\ \mathbf{0} & \infty^{-1}\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix} = \mathbf{P}_1^{-1} + \begin{bmatrix}\mathbf{H}^T & \mathbf{H}_0{}^T\end{bmatrix}\begin{bmatrix}\mathbf{P}_2^{-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}$$

$$= \mathbf{P}_1^{-1} + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H}$$

$$\mathbf{X} = \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{P}_{2A}^{-1}\mathbf{X}_{2A}) = \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H}\begin{bmatrix}\mathbf{H}\\ \mathbf{H}_0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{X}_2\\ \mathbf{0}\end{bmatrix})$$

$$= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{H}^T\mathbf{P}_2^{-1}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{X}_2\\ \mathbf{0}\end{bmatrix}) = \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{X}_2)$$

$$\mathbf{P}_i = \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{P}_{2A}^{-1}\mathbf{P}_{2Ai}\mathbf{P}_{2A}^{-1})\mathbf{P}$$

$$= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{H}^T\mathbf{P}_2^{-1}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{P}_{2i} & \mathbf{0}\\ \mathbf{0} & \infty\end{bmatrix}\begin{bmatrix}\mathbf{I}\\ \mathbf{0}\end{bmatrix}\mathbf{P}_2^{-1}\mathbf{H})\mathbf{P}$$

$$= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{P}_{2i}\mathbf{P}_2^{-1}\mathbf{H})\mathbf{P}$$

$$\mathbf{P}_d = \mathbf{P} - \mathbf{P}_i$$

So we have (3-2). End. □

The split covariance intersection filter can also be given as in (3-3):

$$\begin{aligned}
\mathbf{P}_1 &= \mathbf{P}_{1d} / w + \mathbf{P}_{1i} \\
\mathbf{P}_2 &= \mathbf{P}_{2d} /(1-w) + \mathbf{P}_{2i} \\
\mathbf{K} &= \mathbf{P}_1\mathbf{H}^T(\mathbf{H}\mathbf{P}_1\mathbf{H}^T + \mathbf{P}_2)^{-1} \\
\mathbf{X} &= \mathbf{X}_1 + \mathbf{K}(\mathbf{X}_2 - \mathbf{H}\mathbf{X}_1) \\
\mathbf{P} &= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_1 \\
\mathbf{P}_i &= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_{1i}(\mathbf{I} - \mathbf{K}\mathbf{H})^T + \mathbf{K}\mathbf{P}_{2i}\mathbf{K}^T \\
\mathbf{P}_d &= \mathbf{P} - \mathbf{P}_i
\end{aligned} \tag{3-3}$$

Proof: From (3-2) we have:

$$\begin{aligned}
\mathbf{P} &= (\mathbf{P}_1^{-1} + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H})^{-1} = [\mathbf{P}_1^{-1}(\mathbf{I} + \mathbf{P}_1\mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H})]^{-1} \\
&= (\mathbf{I} + \mathbf{P}_1\mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H})^{-1}\mathbf{P}_1 = [\textstyle\sum_{i=0}^{\infty}(-\mathbf{P}_1\mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{H})^i]\mathbf{P}_1 \\
&= \{\mathbf{I} - \mathbf{P}_1\mathbf{H}^T[\textstyle\sum_{i=0}^{\infty}(-\mathbf{P}_2^{-1}\mathbf{H}\mathbf{P}_1\mathbf{H}^T)^i]\mathbf{P}_2^{-1}\mathbf{H}\}\mathbf{P}_1 \\
&= [\mathbf{I} - \mathbf{P}_1\mathbf{H}^T(\mathbf{I} + \mathbf{P}_2^{-1}\mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1}\mathbf{P}_2^{-1}\mathbf{H}]\mathbf{P}_1 \\
&= [\mathbf{I} - \mathbf{P}_1\mathbf{H}^T(\mathbf{P}_2 + \mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1}\mathbf{H}]\mathbf{P}_1 \\
&= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_1
\end{aligned}$$

$$\begin{aligned}
\mathbf{X} &= \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{X}_2) = \mathbf{P}\mathbf{P}_1^{-1}\mathbf{X}_1 + \mathbf{P}\mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{X}_2 \\
&= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{X}_1 + \mathbf{P}\mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{X}_2
\end{aligned}$$

Where

$$\begin{aligned}
\mathbf{P}\mathbf{H}^T\mathbf{P}_2^{-1} &= [\mathbf{I} - \mathbf{P}_1\mathbf{H}^T(\mathbf{P}_2 + \mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1}\mathbf{H}]\mathbf{P}_1\mathbf{H}^T\mathbf{P}_2^{-1} \\
&= \mathbf{P}_1\mathbf{H}^T[\mathbf{I} - (\mathbf{P}_2 + \mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{P}_1\mathbf{H}^T]\mathbf{P}_2^{-1} \\
&= \mathbf{P}_1\mathbf{H}^T(\mathbf{P}_2 + \mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1}\mathbf{P}_2\mathbf{P}_2^{-1} = \mathbf{P}_1\mathbf{H}^T(\mathbf{P}_2 + \mathbf{H}\mathbf{P}_1\mathbf{H}^T)^{-1} = \mathbf{K}
\end{aligned}$$

Therefore

$$\mathbf{X} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{X}_1 + \mathbf{K}\mathbf{X}_2 = \mathbf{X}_1 + \mathbf{K}(\mathbf{X}_2 - \mathbf{H}\mathbf{X}_1)$$

$$\mathbf{P}_i = \mathbf{P}(\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1} + \mathbf{H}^T\mathbf{P}_2^{-1}\mathbf{P}_{2i}\mathbf{P}_2^{-1}\mathbf{H})\mathbf{P}$$
$$= [(\mathbf{I}-\mathbf{KH})\mathbf{P}_1](\mathbf{P}_1^{-1}\mathbf{P}_{1i}\mathbf{P}_1^{-1})[(\mathbf{I}-\mathbf{KH})\mathbf{P}_1]^T + (\mathbf{PH}^T\mathbf{P}_2^{-1})\mathbf{P}_{2i}(\mathbf{PH}^T\mathbf{P}_2^{-1})^T$$
$$= (\mathbf{I}-\mathbf{KH})\mathbf{P}_{1i}(\mathbf{I}-\mathbf{KH})^T + \mathbf{KP}_{2i}\mathbf{K}^T$$
$$\mathbf{P}_d = \mathbf{P} - \mathbf{P}_i$$

End. □

In (3-3), let the $\mathbf{P}_{1d}$ and $\mathbf{P}_{2d}$ be zero and the split covariance intersection filter (3-3) will become the same to the Kalman filter of partial observation form.

3.3 Basic Functionalities for Cooperative Localization

Suppose there are multiple vehicles and the total number of vehicles is unknown— the cooperative localization is realized in decentralized manner: each vehicle can only and need only to interact (perceiving and communicating) with its neighbouring vehicles. The following functionalities are assumed available; they are abstracted from field practice based on their feasibility in reality.

Absolute positioning function: each vehicle is able to obtain a measurement on its position in an absolute reference or global reference. Absolute positioning function can be realized in different ways; a popular way is to use GPS to provide global positioning measurements.

Relative positioning function: each vehicle is able to measure the relative position of neighbouring vehicles (vehicle-to-vehicle relative pose estimation). Here, *neighbour* indicates being within the perception range. In reality, perceptive sensors such as laser scanner can realize relative positioning function.

Motion monitoring function: each vehicle is equipped with motion sensors which output measurements on its motion state (longitudinal motion and lateral motion). Motion data can be provided by sensors such as odometers (including steering encoders), accelerometers, gyroscopes etc.

Communication function: data can be shared among neighbouring vehicles. Effective communication range is usually larger than effective perception range; so the term *neighbour* mentioned above has an additional meaning of being able to communicate with.

Time-stamping function: the vehicles are able to timestamp their data according to an absolute time reference. In reality, the system time of each intelligent vehicle can be related to the GPS universal time; even a low cost GPS can provide timing information with fairly high precision.

3.4 Cooperative Localization Using Split Covariance Intersection Filter

The idea of using split covariance intersection filter for cooperative localization in the context of intelligent vehicle systems was introduced in our previous works [Li & Nashashibi 2012a]. However, the architecture presented originally in [Li & Nashashibi 2012a] has two implicit requirements: first, direct vehicle-to-vehicle relative pose estimation (see Section 2.4.3) is performed during cooperative localization; second, vehicle localization (vehicle pose estimation) is entangled with detected object tracking (detected object velocity estimation). In fact, these two requirements are NOT NECESSARY for cooperative localization. Therefore, we improve this original architecture and make it more general.

3.4.1 Decomposed Group State

The proposed method is of decentralized architecture; the distributed formalism for each vehicle is the same; therefore, the formalism will be described just from the perspective of one single vehicle (referred to as *ego-vehicle*). The ego-vehicle maintains an estimate of its group state $\mathbf{X}_G$:

$$\mathbf{X}_G = \{\mathbf{X}_E; \mathbf{X}_L\}$$

The group state $\mathbf{X}_G$ consists of two parts: the $\mathbf{X}_E=[x_e;y_e;\theta_e]^T$ denotes the pose of the ego-vehicle; the $\mathbf{X}_L$ is used to generally denote the estimate for its local neighboring environment. The (x,y) and θ respectively denote the position and the heading angle of the vehicle in the absolute reference.

The term *group state* here is similar to but is different from that in [Karam *et al.* 2006b]: the group state in our method (only one estimate for the group state) can be updated with the estimates sent from other vehicles and can also be shared with other vehicles. In [Karam *et al.* 2006b], each vehicle maintains two estimates for its group state: one

estimate, which is updated only using the sensor data of the ego-vehicle, can be shared with other vehicles but can not be updated with the estimates sent from other vehicles; the other one can be updated with the estimates sent from other vehicles but can not be further shared with other vehicles.

There is another difference: in [Karam *et al.* 2006b], a large global covariance matrix is maintained for the whole group state; on the other hand, in our method, the covariance matrix for the ego-vehicle state is separated from the estimate for the neighboring environment. In other words, the group state in our method is a decomposed group state, denoted as:

$$\begin{aligned}\mathbf{E}_G &= \{\mathbf{X}_G, \mathbf{P}_G\} = \{\{\mathbf{X}_E, \mathbf{P}_E\}; \mathbf{X}_L\} \\ &= \{\{\mathbf{X}_E, \mathbf{P}_{dE} + \mathbf{P}_{iE}\}; \mathbf{X}_L\}\end{aligned}$$

where the subscript d and i denote respectively the correlated component and the independent component of the covariance.

3.4.2 State Evolution

The motion of the vehicles can be modelled according to kinematic bicycle model, as illustrated in Figure 3.1:

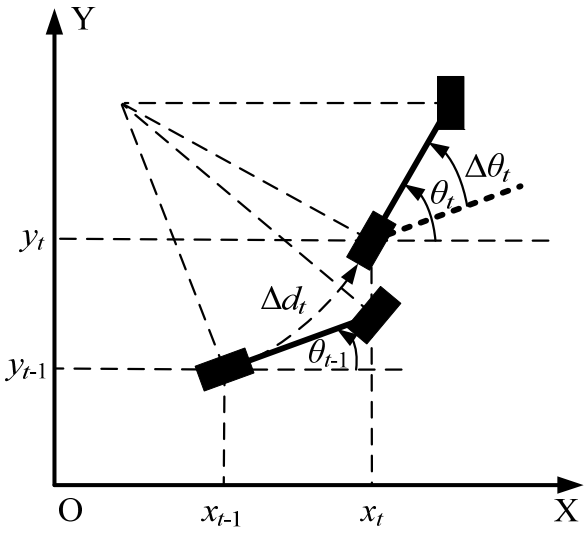


Figure 3.1 Kinematic bicycle model

$$\begin{cases} x_t \approx x_{t-1} + \Delta d_t \cos(\theta_{t-1} + \Delta\theta_t / 2) \\ y_t \approx y_{t-1} + \Delta d_t \sin(\theta_{t-1} + \Delta\theta_t / 2) \\ \theta_t = \theta_{t-1} + \Delta\theta_t \end{cases} \tag{3-4}$$

Proof: Concerning vehicle motion during a short period, we assume that vehicle velocity and vehicle yawrate are both constant:

$$\begin{cases} dx / dt = v\cos(\theta) \\ dy / dt = v\sin(\theta) \\ d\theta / dt = w \end{cases}$$

Suppose w is non-zero; we make integration on x component:

$$x_t = x_{t-1} + \int_{T_{t-1}}^{T_t} v\cos\theta dt = x_{t-1} + v\int_{T_{t-1}}^{T_t} \cos\theta d\theta \frac{dt}{d\theta}$$

$$= x_{t-1} + v\frac{\sin\theta_t - \sin\theta_{t-1}}{w} = x_{t-1} + v\frac{2}{w}\sin(\frac{\theta_t - \theta_{t-1}}{2})\cos(\frac{\theta_t + \theta_{t-1}}{2})$$

$$\approx x_{t-1} + v(\frac{\theta_t - \theta_{t-1}}{w})\cos(\frac{\theta_t + \theta_{t-1}}{2}) = x_{t-1} + v(T_t - T_{t-1})\cos(\frac{\theta_t + \theta_{t-1}}{2})$$

$$= x_{t-1} + \Delta d_t \cos(\theta_{t-1} + \Delta\theta_t / 2)$$

We make integration on y component:

$$y_t = y_{t-1} + \int_{T_{t-1}}^{T_t} v\sin\theta dt = y_{t-1} + v\int_{T_{t-1}}^{T_t} \sin\theta d\theta \frac{dt}{d\theta}$$

$$= y_{t-1} + v\frac{\cos\theta_{t-1} - \cos\theta_t}{w} = y_{t-1} + v\frac{2}{w}\sin(\frac{\theta_t - \theta_{t-1}}{2})\sin(\frac{\theta_t + \theta_{t-1}}{2})$$

$$\approx y_{t-1} + v(\frac{\theta_t - \theta_{t-1}}{w})\sin(\frac{\theta_t + \theta_{t-1}}{2}) = y_{t-1} + v(T_t - T_{t-1})\sin(\frac{\theta_t + \theta_{t-1}}{2})$$

$$= y_{t-1} + \Delta d_t \sin(\theta_{t-1} + \Delta\theta_t / 2)$$

If w is zero, we can verify that (3-4) also holds. End. □

The motion formula (3-4) is denoted in compact form by the function G, as follows:

$$\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} = G(\begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \theta_{t-1} \end{bmatrix}, \begin{bmatrix} \Delta d_t \\ \Delta \theta_t \end{bmatrix})$$

$$\mathbf{X}_t = G(\mathbf{X}_{t-1}, \mathbf{u}_t)$$

The state of the ego-vehicle $\mathbf{X}_{E(t)}$ can be evolved using its motion measurements $\mathbf{u}_{E(t)}$ (here, it can be the original motion sensor measurements, or can be the data corrected by perceptive sensors such as laser scanner).

$$\mathbf{X}_{E(t)} = G(\mathbf{X}_{E(t-1)}, \mathbf{u}_{E(t)}) \tag{3-5}$$

The $\mathbf{u}_{E(t)}$ is assumed to follow the Gaussian distribution $N(\mathbf{0},\mathbf{\Sigma}_u)$. The evolution of the ego-vehicle state covariance is given as:

$$\begin{aligned} \mathbf{P}_{E(t)} &= \mathbf{G}_{\mathbf{X}e}\mathbf{P}_{E(t-1)}\mathbf{G}_{\mathbf{X}e}{}^T + \mathbf{G}_{\mathbf{u}}\mathbf{\Sigma}_{\mathbf{u}}\mathbf{G}_{\mathbf{u}}{}^T + \mathbf{R}_{E(t)} \\ \mathbf{P}_{iE(t)} &= \mathbf{G}_{\mathbf{X}e}\mathbf{P}_{iE(t-1)}\mathbf{G}_{\mathbf{X}e}{}^T + \mathbf{G}_{\mathbf{u}}\mathbf{\Sigma}_{\mathbf{u}}\mathbf{G}_{\mathbf{u}}{}^T + \mathbf{R}_{iE(t)} \end{aligned} \tag{3-6}$$

The $\mathbf{G}_{Xe}$ and $\mathbf{G}_u$ are respectively the Jacobian matrices of the function G with respect to the $\mathbf{X}_E$ and $\mathbf{u}_E$; the $\mathbf{R}_{E(t)}$ and $\mathbf{R}_{iE(t)}$ characterize the motion model error. Since the split covariance intersection filter is used, not only the total covariance $\mathbf{P}_{E(t)}$ but also the independent part $\mathbf{P}_{iE(t)}$ are evolved.

$$\mathbf{G}_{\mathbf{X}e} = \frac{\partial G(\mathbf{X}_{t-1}, \mathbf{u}_t)}{\partial \mathbf{X}_{t-1}} = \begin{bmatrix} 1 & 0 & -\Delta d_t \sin(\theta_{t-1} + \Delta\theta_t / 2) \\ 0 & 1 & \Delta d_t \cos(\theta_{t-1} + \Delta\theta_t / 2) \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{G}_{\mathbf{u}} = \frac{\partial G(\mathbf{X}_{t-1}, \mathbf{u}_t)}{\partial \mathbf{u}_t} = \begin{bmatrix} \cos(\theta_{t-1} + \Delta\theta_t / 2) & -\Delta d_t \sin(\theta_{t-1} + \Delta\theta_t / 2) / 2 \\ \sin(\theta_{t-1} + \Delta\theta_t / 2) & \Delta d_t \cos(\theta_{t-1} + \Delta\theta_t / 2) / 2 \\ 0 & 1 \end{bmatrix}$$

We can assume Δd_t and $\Delta\theta_t$ are independent of each other, then we can set $\mathbf{\Sigma}_u$ to be the diagonal matrix:

$$\mathbf{\Sigma_u} = \begin{bmatrix} \sigma_{\Delta dt} & 0 \\ 0 & \sigma_{\Delta \theta t} \end{bmatrix}$$

The traversed distance Δd_t and the yaw change $\Delta \theta_t$ are respectively the integral of vehicle (longitudinal) velocity v_t and vehicle yawrate w_t:

$$\Delta d_t = \int_{T_{t-1}}^{T_t} v_t dt$$
$$\Delta \theta_t = \int_{T_{t-1}}^{T_t} w_t dt$$

If we assume that the errors of vehicle velocity and vehicle yawrate follow Gaussian white noise distribution, then the diagonal elements of $\mathbf{\Sigma}_u$ are proportional to the period ΔT or T_t-T_{t-1} i.e.

$$\sigma_{\Delta dt} = E[(\int_{T_{t-1}}^{T_t} \tilde{v}_t dt)(\int_{T_{t-1}}^{T_t} \tilde{v}_t dt)^T] = \Delta T \cdot E[\tilde{v}_t \tilde{v}_t{}^T] = \Delta T \cdot \sigma_v$$
$$\sigma_{\Delta \theta t} = E[(\int_{T_{t-1}}^{T_t} \tilde{w}_t dt)(\int_{T_{t-1}}^{T_t} \tilde{w}_t dt)^T] = \Delta T \cdot E[\tilde{w}_t \tilde{w}_t{}^T] = \Delta T \cdot \sigma_w$$

The σ_v and σ_w are constants and set empirically.

3.4.3 State Update with Absolute Positioning Measurements

Let the absolute positioning measurement for the ego-vehicle be denoted as $\mathbf{Z}_A=(x_A,y_A)$. The measurement model can be described as (at time t):

$$\mathbf{Z}_{A(t)} = \mathbf{H}_{A(2\times3)} \mathbf{X}_{E(t)} + \mathbf{R}_A$$
$$\mathbf{H}_{A(2\times3)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

The measurement error $\mathbf{R}_A$ is assumed to follow the Gaussian distribution $N(\mathbf{0},\mathbf{\Sigma}_A)$. Notice that the absolute positioning measurement is completely independent of any existing estimates or any other measurements, the split covariance intersection filter is carried out as follows:

$$\mathbf{K} = \mathbf{P}_{E(t)} \mathbf{H}_A{}^T (\mathbf{H}_A \mathbf{P}_{E(t)} \mathbf{H}_A{}^T + \mathbf{\Sigma}_A)^{-1}$$

$$
\begin{aligned}
\mathbf{X}_{E(t)} &= \mathbf{X}_{E(t)} + \mathbf{K}(\mathbf{Z}_{A(t)} - \mathbf{H}_A \mathbf{X}_{E(t)}) \\
\mathbf{P}_{E(t)} &= (\mathbf{I} - \mathbf{K}\mathbf{H}_A)\mathbf{P}_{E(t)} \\
\mathbf{P}_{iE(t)} &= (\mathbf{I} - \mathbf{K}\mathbf{H}_A)\mathbf{P}_{iE(t)}(\mathbf{I} - \mathbf{K}\mathbf{H}_A)^T + \mathbf{K}\boldsymbol{\Sigma}_A\mathbf{K}^T \\
\mathbf{P}_{dE(t)} &= \mathbf{P}_{E(t)} - \mathbf{P}_{iE(t)}
\end{aligned}
\tag{3-7}
$$

3.4.4 State Update with Relative Positioning Measurements and the States of Other Vehicles

The ego-vehicle can share its group state with its neighboring vehicles via inter-vehicle communication; vice versa, it can also receive the group state estimates of its neighboring vehicles. Without loss of generality, let the group state shared by a neighboring vehicle (referred to as *communicated ego-vehicle*) be denoted as:

$$
\begin{aligned}
\mathbf{E}_{CG} &= \{\mathbf{X}_{CG}, \mathbf{P}_{CG}\} = \{\{\mathbf{X}_{CE}, \mathbf{P}_{CE}\}; \mathbf{X}_{CL}\} \\
&= \{\{\mathbf{X}_{CE}, \mathbf{P}_{dCE} + \mathbf{P}_{iCE}\}; \mathbf{X}_{CL}\}
\end{aligned}
$$

The $\mathbf{X}_{CE}=[x_{ce};y_{ce};\theta_{ce}]^T$ denotes the state of the communicated ego-vehicle and the $\mathbf{X}_{CL}$ generally denotes its estimate for its local neighboring environment. The subscript *d* and *i* denote respectively the correlated component and the independent component of the covariance.

According to the relative positioning function assumption, the relative pose between the ego-vehicle and the communicated ego-vehicle can be obtained. This relative pose can be revealed via analysis of the $\mathbf{X}_L$ and $\mathbf{X}_{CL}$. For example, if the ego-vehicle can directly detect the entire communicated ego-vehicle and succeed in associating the latter, then the relative pose can be revealed from $\mathbf{X}_L$; if the communicated ego-vehicle can directly detect the entire ego-vehicle and succeed in associating the latter, then the relative pose can be revealed from $\mathbf{X}_{CL}$. We can generally treat the relative pose between the ego-vehicle and the communicated ego-vehicle (denoted as $\mathbf{Z}_R$) as the output of a functional *R* in terms of $\mathbf{X}_L$ and $\mathbf{X}_{CL}$ i.e.

$$
\mathbf{Z}_R = \begin{bmatrix} \Delta x_R \\ \Delta y_R \\ \Delta \theta_R \end{bmatrix} = R(\mathbf{X}_L, \mathbf{X}_{CL})
$$

The error that affects $\mathbf{Z}_R$ is assumed to follow Gaussian distribution $N(\mathbf{0},\mathbf{\Sigma}_R)$. The state of the ego-vehicle can be indirectly computed by compounding the relative pose estimate and the state estimate of the communicated ego-vehicle; this indirect estimate of the ego-vehicle state is denoted as $\mathbf{X}_{EI}$ with covariance $\mathbf{P}_{EI}$. (Refer to the compounding operation in Appendix I)

$$\mathbf{X}_{EI} = \mathbf{X}_{CE} \oplus \mathbf{Z}_R = \begin{bmatrix} x_{ce} \\ y_{ce} \\ \theta_{ce} \end{bmatrix} \oplus \begin{bmatrix} \Delta x_R \\ \Delta y_R \\ \Delta\theta_R \end{bmatrix} = \begin{bmatrix} \Delta x_R \cos\theta_{ce} - \Delta y_R \sin\theta_{ce} + x_{ce} \\ \Delta x_R \sin\theta_{ce} + \Delta y_R \cos\theta_{ce} + y_{ce} \\ \Delta\theta_R + \theta_{ce} \end{bmatrix}$$

$$\mathbf{P}_{EI} \approx (\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{X}_{CE}})\mathbf{P}_{CE}(\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{X}_{CE}})^T + (\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{Z}_R})\mathbf{\Sigma}_R(\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{Z}_R})^T$$

$$\mathbf{P}_{iEI} \approx (\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{X}_{CE}})\mathbf{P}_{iCE}(\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{X}_{CE}})^T + (\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{Z}_R})\mathbf{\Sigma}_{iR}(\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{Z}_R})^T$$

The $\partial\mathbf{X}_{EI}/\partial\mathbf{X}_{CE}$ and $\partial\mathbf{X}_{EI}/\partial\mathbf{Z}_R$ are respectively the Jacobian matrices of $\mathbf{X}_{EI}$ with respect to the $\mathbf{X}_{CE}$ and $\mathbf{Z}_R$.

$$\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{X}_{CE}} = \begin{bmatrix} 1 & 0 & -\Delta x_R \sin\theta_{ce} - \Delta y_R \cos\theta_{ce} \\ 0 & 1 & \Delta x_R \cos\theta_{ce} - \Delta y_R \sin\theta_{ce} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial \mathbf{X}_{EI}}{\partial \mathbf{Z}_R} = \begin{bmatrix} \cos\theta_{ce} & -\sin\theta_{ce} & 0 \\ \sin\theta_{ce} & \cos\theta_{ce} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Then $\mathbf{X}_E$ is fused with $\mathbf{X}_{EI}$ using the split covariance intersection filter as follows:

$$\mathbf{P}_1 = \mathbf{P}_{dE(t)} / w + \mathbf{P}_{iE(t)}$$

$$\mathbf{P}_2 = \mathbf{P}_{dEI(t)} /(1-w) + \mathbf{P}_{iEI(t)}$$

$$
\begin{aligned}
\mathbf{K} &= \mathbf{P}_1(\mathbf{P}_1+\mathbf{P}_2)^{-1} \\
\mathbf{X}_{E(t)} &= \mathbf{X}_{E(t)} + \mathbf{K}(\mathbf{X}_{EI(t)} - \mathbf{X}_{E(t)}) \\
\mathbf{P}_{E(t)} &= (\mathbf{I}-\mathbf{K})\mathbf{P}_1 \\
\mathbf{P}_{iE(t)} &= (\mathbf{I}-\mathbf{K})\mathbf{P}_{iE(t)}(\mathbf{I}-\mathbf{K})^T + \mathbf{K}\mathbf{P}_{iEI(t)}\mathbf{K}^T \\
\mathbf{P}_{dE(t)} &= \mathbf{P}_{E(t)} - \mathbf{P}_{iE(t)}
\end{aligned}
\tag{3-8}
$$

The w is determined by minimizing the determinant of the new covariance. Some practical issues deserve further explanations:

Covariance of the relative pose estimate $\mathbf{Z}_\mathbf{R}$: the covariance $\mathbf{\Sigma}_R$ for $\mathbf{Z}_R$ is also divided into two parts, the correlated part $\mathbf{\Sigma}_{dR}$ and the independent part $\mathbf{\Sigma}_{iR}$. In practice, it might be difficult to have a systematic way to analytically compute the $\mathbf{\Sigma}_R$ for $\mathbf{Z}_R$; the $\mathbf{\Sigma}_R$ has to be set empirically to characterize the uncertainty of the relative pose estimate. The ratio of $\mathbf{\Sigma}_{iR}$ (or $\mathbf{\Sigma}_{dR}$) in $\mathbf{Z}_R$ has also to be set empirically. For simplicity, we can set $\mathbf{\Sigma}_{iR}$ (or $\mathbf{\Sigma}_{dR}$) to be a half of $\mathbf{Z}_R$.

Communication delay: In reality, when the communicated ego-vehicle has a new state estimate and shares it with the ego-vehicle, the ego-vehicle can not use the shared state immediately due to communication delay (including the time needed by the communicated ego-vehicle to package the shared state into data format suitable for transmitting, the time needed to transmit the data, the time needed by the ego-vehicle to unpack the data so that it can use the whole shared state). As vehicular communication technology has developed rapidly, careful implementation can reduce the communication delay to no more than dozens of milliseconds; the communication delay will be further reduced in future. There are two ways to deal with the communication delay. If the error caused by the communication delay makes up a small part of the overall error, then the error can be simply treated as random error. For example, if the communication delay is 20ms and the vehicle velocity is 5m/s, then this communication delay will result in an error of 0.1m which can be fairly treated as random error. In contrast, if the communication delay is 50ms and the vehicle velocity is 20m/s, then this communication delay will result in an error of 1.0m, then the delayed state estimate ($\mathbf{X}_{EI}$ and $\mathbf{P}_{EI}$) had better be compensated by the motion data of the ego-vehicle, as in (3-5) and (3-6).

3.4.5 Cooperative Localization Architecture

Cooperative localization is realized in decentralized (distributed) manner. From the perspective of an intelligent vehicle, the localization procedures are as follows (the procedures flow diagram is illustrated in Figure 3.2):

At each period, the vehicle evolves its state estimate (including covariance) using its motion measurements, according to (3-5) and (3-6).
When the vehicle has absolute positioning measurement of its own, it updates its state estimate according to (3-7).
When the vehicle receives data from a neighbouring vehicle, it updates its state estimate according to (3-8).

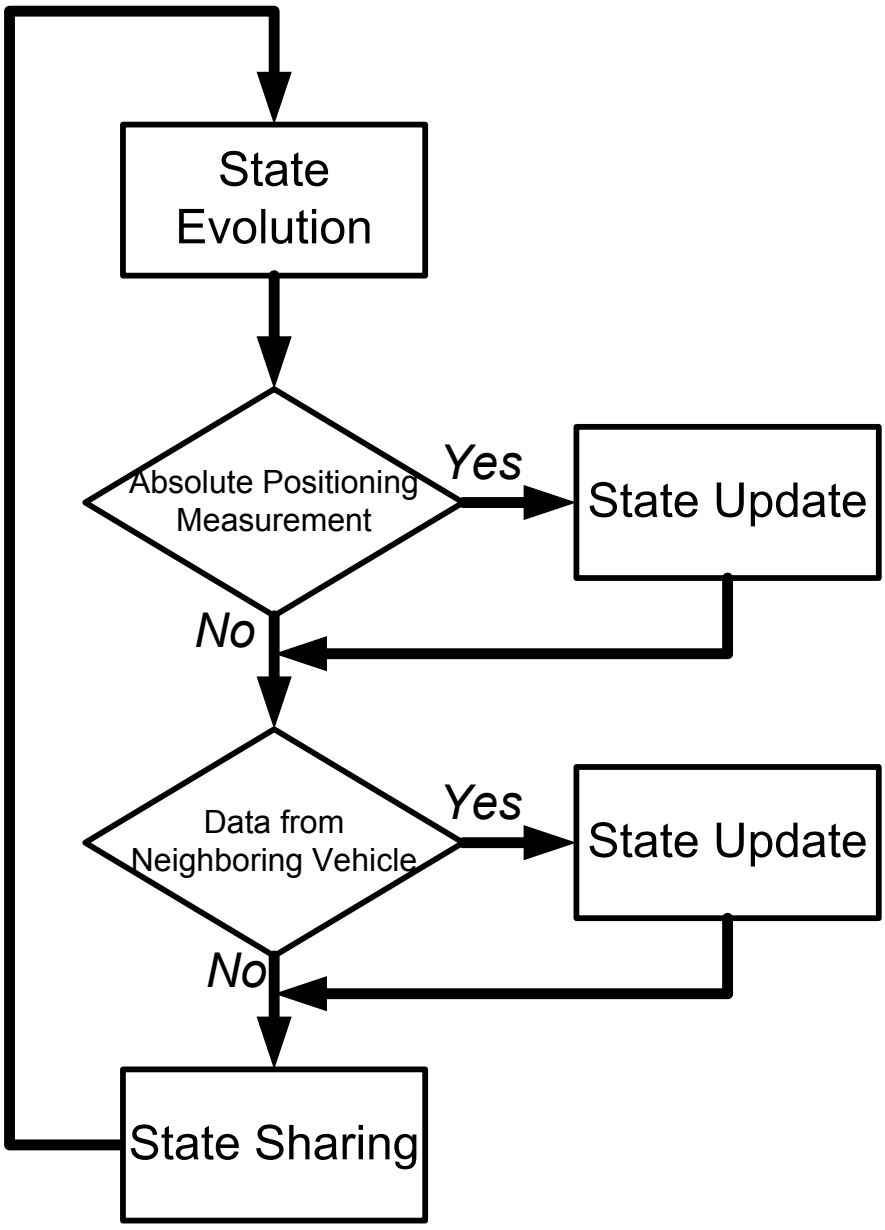


Figure 3.2 Distributed cooperative localization procedures at one vehicle

As we can see, this distributed cooperative localization architecture is rather simple: when the vehicle has some new data from itself or from another vehicle, it can use the new data to evolve or update its state estimate; when the vehicle has new state estimate, it can also share its data with other vehicles; no monitoring or controlling of data flow within vehicle networks is needed. Despite of the simplicity of this architecture, the risk

of over-convergence can be essentially removed, because the risk is removed directly by the split covariance intersection filter during estimates fusion.

3.5 Indirect Vehicle-to-Vehicle Relative Pose Estimation

In above introduced architecture of cooperative localization, the functionality of vehicle-to-vehicle (V2V) relative pose estimation is assumed available. In reality, V2V relative pose estimation is not a trivial issue, as we have discussed in Section 2.4.3. In this sub-section, we describe the indirect vehicle-to-vehicle relative pose estimation strategy.

The basic idea is as follows: each vehicle performs local **SLAM** (Simultaneous Localization and Mapping) [Wang 2004] [Vu 2009]. Here, the purpose of local SLAM is not to build a global environment map, but to build in real-time a dynamic local map around the vehicle. As illustrated in Figure 3.3, when two vehicles A and B are in neighborhood and need to estimate the relative pose between them, this relative pose can be indirectly inferred via a chain of geometric relationships among vehicle A, local map of vehicle A, local map of vehicle B, and vehicle B. The relative pose between vehicle A and its local map can be estimated by SLAM method; the relative pose between vehicle B and its local map can also be estimated by SLAM method; the relative pose between local map of A and local map of B can be estimated by **maps merging** method. Then the relative pose between vehicle A and B can be indirectly estimated by compounding the relative poses among vehicle A, local map of A, local map of B, and vehicle B.

Let the pose of vehicle A in its local map be denoted as $\mathbf{p}_{LA}$; let the pose of vehicle B in its local map be denoted as $\mathbf{p}_{LB}$; let the relative pose between the local map of vehicle B and the local map of vehicle A be denoted as $\mathbf{p}_{BA}$. If the estimates of $\mathbf{p}_{LA}$, $\mathbf{p}_{LB}$, and $\mathbf{p}_{BA}$ are available, then the relative pose between vehicle B and vehicle A can be computed as:

$$\mathbf{p}_{vBA} = inv(\mathbf{p}_{LA}) \oplus \mathbf{p}_{BA} \oplus \mathbf{p}_{LB} \tag{3-9}$$

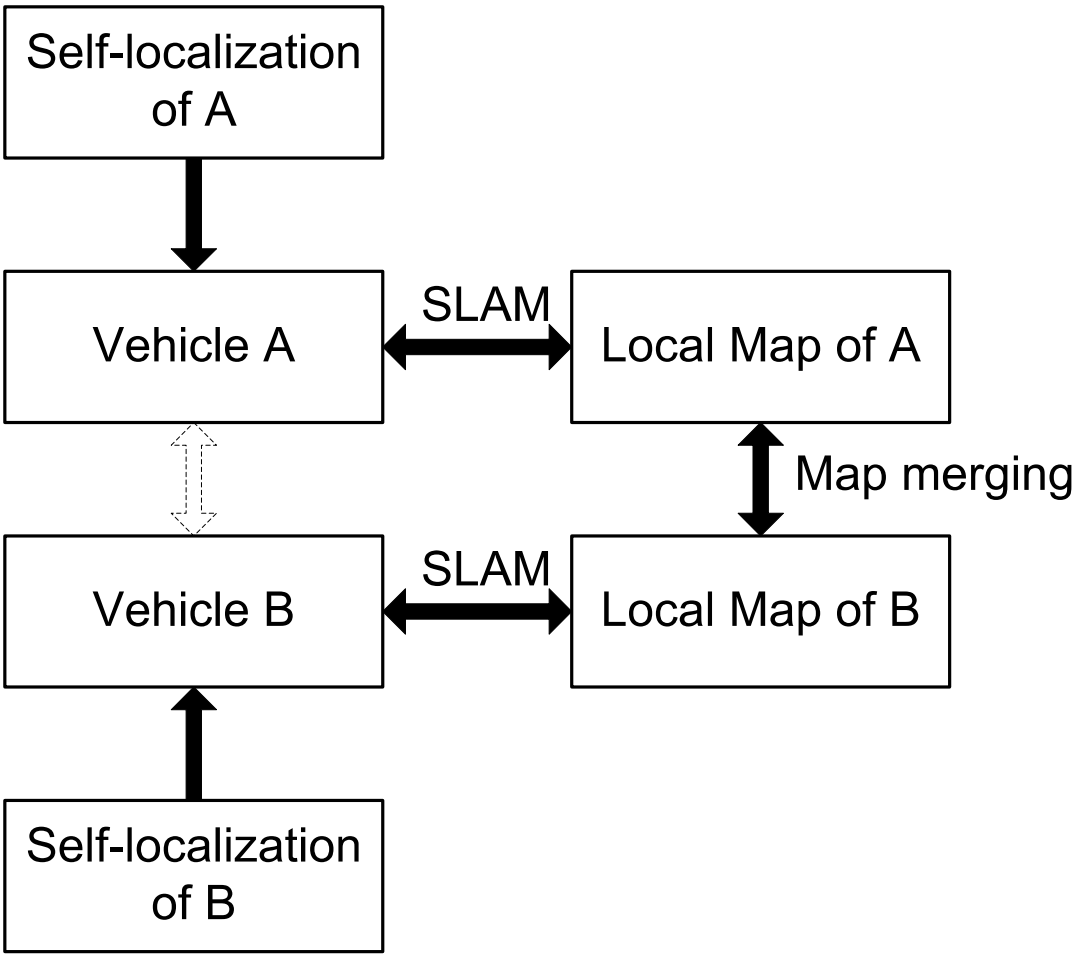


Figure 3.3 Indirect vehicle-to-vehicle relative pose estimation strategy

SLAM suffers from accumulated error, especially when building a large and non-cyclic global map; yet SLAM can achieve desirable accuracy and consistency when building a local map of hundred meters scale [Wang 2004] [Vu 2009]. The local map built on-line is not dedicated only to cooperative localization; it is also valuable for safe navigation (such as collision avoidance) of a single vehicle system. In the indirect V2V relative pose estimation method, the challenging problems of vehicle detection, data association, and relative pose computation in direct V2V relative pose estimation method are implicitly solved during local maps merging (local map of vehicle A and local map of vehicle B as in Figure 3.3). For local maps of a scale such as 80 meters, large and stable objects (buildings, infrastructures etc) are usually the dominating factors, which contribute to successful local maps merging.

Detailed description of the local SLAM method and the maps merging method will be postponed to CHAPTER 4. An example of the local SLAM results and the maps merging results is illustrated in Figure 4.6.

3.6 Summary

We have reviewed the concept of estimate consistency and the split covariance intersection filter; we have presented several forms of this filter together with their derivations and an original proof for the fusion consistency of this filter. We have specified the compounding notation for coordinate transformation and explained some properties of this compounding notation. We have introduced several basic functionalities as the condition for realizing cooperative localization; these functionalities are abstracted from field practice based on their feasibility in reality. We have described a general architecture of cooperative localization using split covariance intersection filter; as the architecture is decentralized, we have described from the perspective of an intelligent vehicle how it can evolve its state estimate using its motion measurements, how it can update its state estimate using its own absolute positioning measurements, and how it can update its state estimate with the data shared by neighbouring vehicles. We have presented the indirect vehicle-to-vehicle relative pose estimation strategy; the concrete realization of some components in this strategy will be detailed in CHAPTER 4.

CHAPTER 4 Cooperative Local Mapping and Moving Objects Detection

Résumé

Nous avons fourni une solution de cartographie locale coopérative pour les véhicules intelligents fondée sur la télémétrie laser. Nous avons décrit la méthode de SLAM local fondée sur la grille d'occupation. Nous faisons la distinction entre état local et état global puis nous décrivons comment les estimations de l'état local du véhicule obtenues par le SLAM peuvent être utilisées pour obtenir les estimations de l'état global de celui-ci. Nous avons présenté l'architecture de fusion des grilles d'occupation en formalisant le problème dans un cadre généralisé de problème d'optimisation. Nous avons proposé une nouvelle fonction objective qui mesure le degré cohérence de l'alignement des cartes fondé sur la probabilité d'occupation. Ensuite, nous avons proposé une approche fondée sur un algorithme génétique dans le but de rechercher l'alignement optimal des grilles. Nous avons enfin introduit l'architecture de la détection coopérative des objets en mouvement, fondé sur la fusion des grilles occupations. Pour une mise en œuvre complète, nous avons adopté deux méthodes de base pour la détection des objets en mouvement.

4.1 Introduction

In this chapter, we describe in details our solution of multi-vehicles cooperative local mapping and moving objects detection for laser scanner based intelligent vehicles. The method architecture is as follows: each vehicle establishes in real-time a local occupancy grid map and performs moving objects detection based on the established occupancy grid map. During vehicles cooperation, the local occupancy grid maps of different vehicles are merged, so that these different vehicles can be spatially related to each other; then the moving objects detection results of these vehicles can also be merged. First, we describe the method of occupancy grid based single vehicle local SLAM (Simultaneous Localization And Mapping) and how local SLAM results can be used to assist vehicle global localization. Next, we introduce a new method of occupancy grid maps merging, which consists in a new objective function that measures the consistency degree of maps alignment and a genetic algorithm that searches for the optimal maps alignment. Then, based on the proposed occupancy grid maps merging method, we introduce the scheme of multi-vehicles cooperative moving objects detection.

4.2 Occupancy Grid based Local SLAM

4.2.1 Occupancy Grid based Mapping with Known Vehicle States

For our research context, we prefer the occupancy grid based environment representation for its ability to represent general unstructured outdoor environment. The occupancy grid [Elfes 1989] is a two-dimensional lattice which divides the environment space into rectangular cells; each cell is associated with a real value in the unit interval [0, 1], which is called **occupancy state**. The cell value or the occupancy state of the cell represents the degree of the cell being occupied by or free of object. The cell value 0.5 represents the cell being in unknown state, neither occupied nor free. For cell value larger than 0.5, the larger the cell value is, the more likely the cell is occupied. For cell value smaller than 0.5, the smaller the cell value is, the more likely the cell is free.

Let $\mathbf{S}=[x_s,y_s,\theta_s]^T$ denotes the vehicle local state (or pose) as in local SLAM (**Note**: in order to be different from the denotation for vehicle global state **X**; explanations will be given later. In this sub-section, *vehicle local state* is also generally referred to as *vehicle state* without *local*); **M** denotes the occupancy grid map, and *m* denotes a generic cell in

the occupancy grid map; **u** denotes vehicle motion data; **z** denotes range measurements that are used to build the environment map, and subscript *t* denotes the time index.
Suppose the occupancy states of grid cells are independent of each other and suppose vehicle poses are known. The purpose of the occupancy grid based mapping is to estimate the posterior probability of the occupancy state $p(m|\mathbf{S}_{1:t}, \mathbf{z}_{1:t})$ for each cell *m*. Since we focus on real-time perception, we can use the Bayes rule to derive a recursive scheme to estimate $p(m|\mathbf{S}_{1:t}, \mathbf{z}_{1:t})$, as in Section 2.3.3.

$$p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1}, m)\, p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})}{p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})}$$

Following the Markov assumption, we can treat $\mathbf{z}_t$ as being independent of past vehicle states $\mathbf{S}_{1:t-1}$ and past range measurements $\mathbf{z}_{1:t-1}$. Then, we simplify $p(\mathbf{z}_t|\mathbf{S}_{1:t},\mathbf{z}_{1:t-1},m)$ and use the Bayes rule again:

$$\begin{aligned} p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1}, m) &= p(\mathbf{z}_t \mid \mathbf{S}_t, m) \\ &= \frac{p(m \mid \mathbf{S}_t, \mathbf{z}_t)\, p(\mathbf{z}_t \mid \mathbf{S}_t)}{p(m \mid \mathbf{S}_t)} \end{aligned}$$

Then we have:

$$\begin{aligned} p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) &= \frac{p(m \mid \mathbf{S}_t, \mathbf{z}_t)\, p(\mathbf{z}_t \mid \mathbf{S}_t)}{p(m \mid \mathbf{S}_t)} \frac{p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})}{p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})} \\ &= \frac{p(m \mid \mathbf{S}_t, \mathbf{z}_t)\, p(\mathbf{z}_t \mid \mathbf{S}_t)}{p(m)} \frac{p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}{p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})} \end{aligned} \tag{4-1}$$

During above derivation, the assumption that the *m* and the $\mathbf{S}_t$ is mutually independent has been used, because they are mutually non-generative, as will be illustrated latter in the Dynamic Bayesian Network chart of SLAM in Figure 4.3. Equation (4-1) computes the probability of a cell being occupied; in similar way, we can derive a formula to compute the probability of a cell being free (where $\overline{m}$ denotes a free cell in contrast with *m* that denotes an occupied cell):

$$p(\overline{m} \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = \frac{p(\overline{m} \mid \mathbf{S}_t, \mathbf{z}_t)\, p(\mathbf{z}_t \mid \mathbf{S}_t)}{p(\overline{m})} \frac{p(\overline{m} \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}{p(\mathbf{z}_t \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t-1})} \tag{4-2}$$

Divide (4-1) by (4-2) and we have:

$$\frac{p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t})}{p(\overline{m} \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t})} = \frac{p(m \mid \mathbf{S}_t, \mathbf{z}_t)}{p(\overline{m} \mid \mathbf{S}_t, \mathbf{z}_t)} \frac{p(\overline{m})}{p(m)} \frac{p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}{p(\overline{m} \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})} \tag{4-3}$$

Notice that the probabilities for m and $\overline{m}$ are mutually complementing to 1. We define the *Odds* function as:

$$Odds(x) = \frac{p(x)}{p(\overline{x})} = \frac{p(x)}{1 - p(x)}$$

Then (4-3) turns into:

$$Odds(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = Odds(m \mid \mathbf{S}_t, \mathbf{z}_t) \cdot Odds(m)^{-1} \cdot Odds(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1}) \tag{4-4}$$

The *a priori* probability of m can be fairly assumed to be 0.5 (unknown state); then (4-4) turns into:

$$Odds(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = Odds(m \mid \mathbf{S}_t, \mathbf{z}_t) \cdot Odds(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1}) \tag{4-5}$$

We can easily recover the probability from the *Odds* function:

$$p(x) = \frac{Odds(x)}{1 + Odds(x)} = \frac{1}{1 + Odds(x)^{-1}}$$

From (4-4) we have:

$$p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = [1 + \frac{1 - p(m \mid \mathbf{S}_t, \mathbf{z}_t)}{p(m \mid \mathbf{S}_t, \mathbf{z}_t)} \cdot \frac{p(m)}{1 - p(m)} \cdot \frac{1 - p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}{p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}]^{-1} \tag{4-6}$$

Similarly, when *a priori* probability of m is assumed to be 0.5 (unknown state); then (4-6) turns into:

$$p(m \mid \mathbf{S}_{1:t}, \mathbf{z}_{1:t}) = [1 + \frac{1 - p(m \mid \mathbf{S}_t, \mathbf{z}_t)}{p(m \mid \mathbf{S}_t, \mathbf{z}_t)} \cdot \frac{1 - p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}{p(m \mid \mathbf{S}_{1:t-1}, \mathbf{z}_{1:t-1})}]^{-1} \quad (4\text{-}7)$$

Equation (4-6) and (4-7) give a recursive way to update the old estimate of the occupancy grid map with new vehicle state $\mathbf{S}_t$ and new range measurement $\mathbf{z}_t$.

4.2.2 Inverse Measurement Model

As shown in (4-7), the inverse measurement model $p(m|\mathbf{S}_t,\mathbf{z}_t)$ determines how we can incrementally update the estimate for the occupancy grid map. The *inverse measurement model* is in contrast with *measurement model* $p(\mathbf{z}_t|\mathbf{S}_t,m)$; *inverse* here means the inverse reasoning from the effect to the cause. In reality, the environment is the cause, whereas the measurement is the effect. In other words, we have the environment and then we have the measurement; it is not that we have the measurement and then we have the environment.

In our case, laser scanner is used to provide the range measurement $\mathbf{z}_t$. A frame of range measurement consists of a group of laser beam readings; a laser beam returns the distance measurement of the closest object that it hits along a specified direction with respect to the laser scanner. Denote $\mathbf{z}_t$ as a set of distance measurements:

$$\mathbf{z}_t = \{z_{t,k} \mid 1 \le k \le n\}$$

We assume that each laser beam returns its distance measurement independently of the operation of other laser beams. Then we consider one generic individual laser beam $z_{t,k}$ and consider the one dimensional grid cells along this laser beam, as illustrated in Figure 4.1.

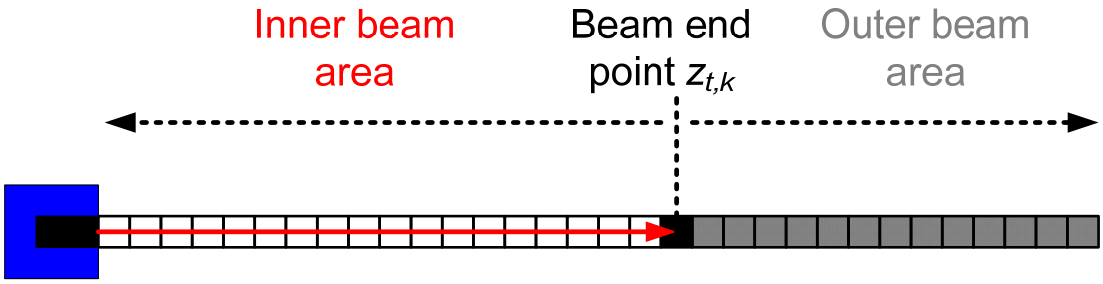


Figure 4.1 A laser beam and the grid cells along it

Suppose the laser scanner is situated at the origin and the laser beam hits an object at distance $z_{t,k}$, indicated by the *beam end point* in Figure 4.1. Suppose the laser beam measurement is ideally reliable and accurate, then we can judge that an object exists at the beam end point and the cell at the beam end point should be occupied. We can also judge that the area before the beam end point i.e. the *inner beam area* is free of any object; otherwise, the laser beam would hit the closer object and return a measurement smaller than $z_{t,k}$. The cells in the inner beam area should be free. For the area after the beam end point i.e. the *outer beam area*, we can not make any judgement whether any object exists there. Concerning a cell in the outer beam area, it could be occupied by an object or not. So the cells in the outer beam area are in unknown state. In summary, we can set the inverse measurement model $p(m|\mathbf{S}_t,z_{t,k})$ ideally as follows:

$$p(m \mid \mathbf{S}_t, z_{t,k}) = \begin{cases} 0 & :\| m - \mathbf{S}_t \| < z_{t,k} \\ 1 & :\| m - \mathbf{S}_t \| = z_{t,k} \\ 0.5 & :\| m - \mathbf{S}_t \| > z_{t,k} \end{cases}$$

In reality, a laser beam measurement has certain degree of errors; besides, we do not want to settle down the judgement based on only one frame of range measurement. Therefore, we set soft thresholds for the inverse measurement model: if a cell is in the inner beam area, we set it to be 80% free (not completely free); if a cell is at the beam end point, we set it to be 80% occupied (not completely occupied). The inverse measurement model is illustrated in Figure 4.2.

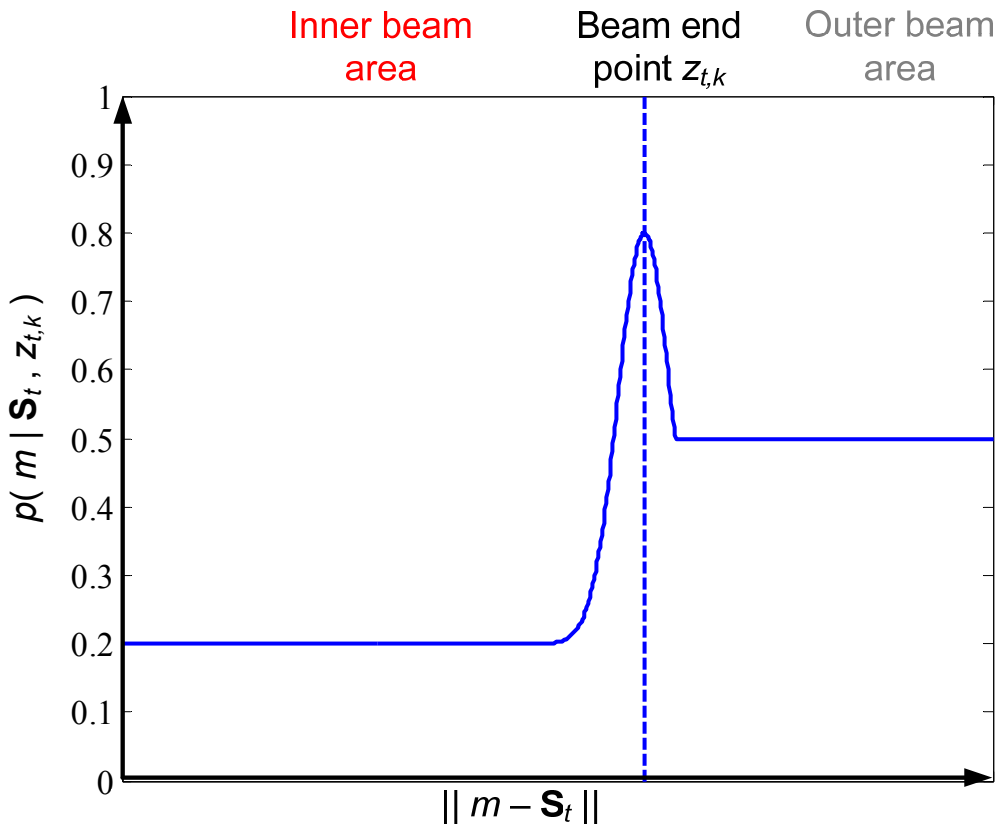


Figure 4.2 The inverse measurement model $p(\boldsymbol{m}|S_t,z_t)$

4.2.3 Incremental Maximum Likelihood SLAM

In Section 4.2.1, we have introduced how to incrementally update the occupancy grid map estimate using range measurements, if vehicle states are already known. In real-time applications, however, the vehicle (local) states themselves are also some states that need to be estimated recursively. The overall process of incremental SLAM can be represented by a Dynamic Bayesian Network graph model as illustrated in Figure 4.3. The vehicle evolves from last state to current state under recent motion actions; range measurements are generated according to the vehicle state and the environment.

We might simply evolve vehicle state according to motion measurements, for example, odometer measurements (including steering measurements) that are commonly used in intelligent vehicle systems. However, odometers suffer from accumulated errors; a navigation distance of only several tens of meters might result in considerable error in vehicle state estimate and apparent inconsistency in the mapping result. We had better

use the inner spatial constraint among the sequence of range measurements to correct odometer based vehicle state prediction. In other words, we rely on the mapping result to obtain more accurate vehicle state estimate. Meanwhile, the mapping process depends on vehicle state estimation. Therefore, we have to carry out a process where vehicle state estimation and environment map estimation are juxtaposed together; this accounts for the meaning of *simultaneous* in SLAM.

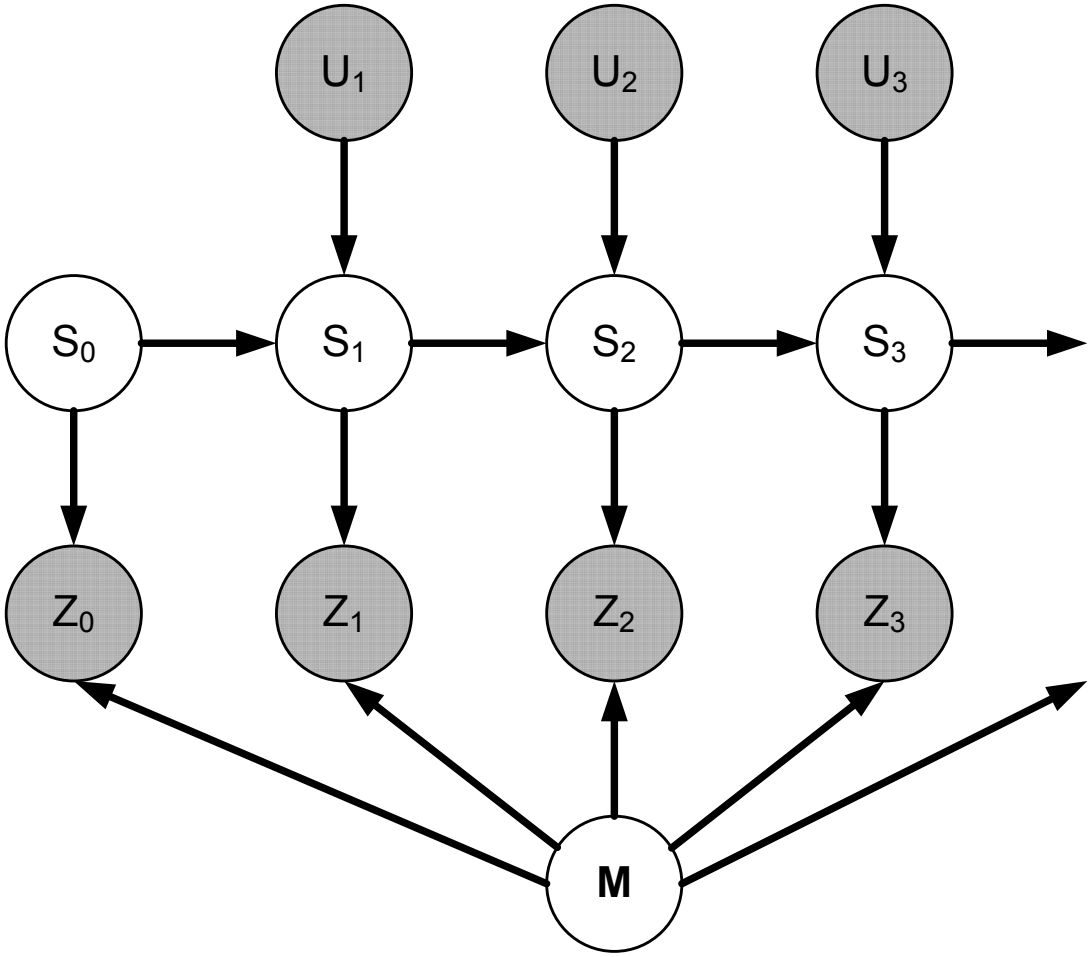


Figure 4.3 Dynamic Bayesian Network of incremental SLAM

The objective of SLAM is to estimate the joint posterior probability distribution $p(\mathbf{S}_t,\mathbf{M}|\mathbf{z}_{0:t},\mathbf{u}_{1:t})$. As in Section 2.3.3, we can use the Bayes rule to derive a recursive estimation scheme for SLAM.

$$
\begin{aligned}
& p(\mathbf{S}_t, \mathbf{M}_t \mid \mathbf{z}_{0:t}, \mathbf{u}_{1:t}) \\
&\quad = \delta p(\mathbf{z}_t \mid \mathbf{S}_t, \mathbf{M}_t, \mathbf{z}_{0:t-1}, \mathbf{u}_{1:t}) p(\mathbf{S}_t, \mathbf{M}_t \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{1:t}) \\
&\quad = \delta p(\mathbf{z}_t \mid \mathbf{S}_t, \mathbf{M}_t) \\
&\quad \int p(\mathbf{S}_t, \mathbf{M}_t \mid \mathbf{S}_{t-1}, \mathbf{M}_{t-1}, \mathbf{z}_{0:t-1}, \mathbf{u}_{1:t}) p(\mathbf{S}_{t-1}, \mathbf{M}_{t-1} \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{1:t}) d(\mathbf{S}_{t-1}, \mathbf{M}_{t-1}) \\
&\quad = \underbrace{\delta p(\mathbf{z}_t \mid \mathbf{S}_t, \mathbf{M}_t)}_{\text{update}} \underbrace{\int p(\mathbf{S}_t \mid \mathbf{S}_{t-1}, \mathbf{u}_t) \underbrace{p(\mathbf{S}_{t-1}, \mathbf{M}_{t-1} \mid \mathbf{z}_{0:t-1}, \mathbf{u}_{1:t-1})}_{\text{posterior at } t-1} d\mathbf{S}_{t-1}}_{\text{evolution (prediction)}}
\end{aligned}
$$

During above derivation, we have used the Markov assumption again and the static map assumption i.e. $\mathbf{M}_t$=$\mathbf{M}_{t\text{-}1}$. Since the dimension of the occupancy grid map is huge, it is difficult to maintain a full posterior distribution estimate during SLAM. We adopt the incremental maximum likelihood estimation framework as introduced in Section 2.3.5; eat each period, we only keep the most likely estimate of vehicle state and the map. More specifically, the incremental maximum likelihood SLAM is formulized as follows:

$$
(\hat{\mathbf{S}}_t, \hat{\mathbf{M}}_t) = \arg\max_{\mathbf{S}_t, \mathbf{M}_t} \{ p(\mathbf{z}_t \mid \mathbf{S}_t, \mathbf{M}_t) p(\mathbf{S}_t, \mathbf{M}_t \mid \hat{\mathbf{S}}_{t-1}, \hat{\mathbf{M}}_{t-1}, \mathbf{u}_t) \} \tag{4-8}
$$

We follow the static map assumption again and then (4-8) turns into:

$$
\hat{\mathbf{S}}_t = \arg\max_{\mathbf{S}_t} \{ p(\mathbf{z}_t \mid \mathbf{S}_t, \hat{\mathbf{M}}_{t-1}) p(\mathbf{S}_t \mid \hat{\mathbf{S}}_{t-1}, \mathbf{u}_t) \} \tag{4-9}
$$

Once the vehicle state $\mathbf{S}_t$ is estimated, we can update the occupancy grid map by appending new data ($\mathbf{S}_t$,$\mathbf{z}_t$) into the old map $\mathbf{M}_{t\text{-}1}$:

$$
\hat{\mathbf{M}}_t = \hat{\mathbf{M}}_{t-1} \bigcup (\hat{\mathbf{S}}_t, \mathbf{z}_t) \tag{4-10}
$$

Incremental maximum likelihood SLAM is a repetition of executing the steps (4-9) and (4-10). In the step (4-10), we can update each grid cell according to (4-7) (For the cells that are not influenced by the new frame of laser beams, we can just leave them unchanged). In the step (4-9), we adopt the occupancy grid based scan matching method as introduced in [Vu 2009] for its computational efficiency as well as its insensitiveness to dynamic entities in the environment. The occupancy grid based scan matching method is briefly reviewed as follows:

Step I: We randomly generate a set of vehicle state samples from last vehicle state estimate and vehicle motion measurements, satisfying the probabilistic vehicle motion model $p(\mathbf{S}_t \mid \hat{\mathbf{S}}_{t-1}, \mathbf{u}_t)$. *More specifically, in each sampling, we generate a vehicle motion sample from the vehicle motion measurement according to a pre-defined motion error model. From this vehicle motion sample and last vehicle state estimate, we can generate a vehicle state sample according to the kinematic bicycle model specified in (3-4). Let the set of vehicle state samples be denoted as:*

$$\mathbf{S}_{t,samples} = \{\mathbf{S}_{t,j} \mid 1 \le j \le sample\_N\}$$

Step II: With a vehicle state sample, we can localize the end point cells of the laser beams in the occupancy map $\boldsymbol{M}_{t-1}$. *The sum of the occupancy states of all the occupied end point cells is used to measure the likelihood value of this vehicle state sample i.e.:*

$$p(\mathbf{z}_t \mid \mathbf{S}_{t,j}, \hat{\mathbf{M}}_{t-1}) = \sum_{k=1}^{n} \{p(\mathbf{S}_{t,j} \oplus z_{t,k}) \mid \mathbf{S}_{t,j} \oplus z_{t,k} \text{ is occupied}\}$$

We select the vehicle state sample with highest likelihood value computed above way and treat it as the solution for the optimization problem in the step (4-9).

4.2.4 Vehicle Local State vs. Vehicle Global State

By so far, the incremental maximum likelihood SLAM scheme presented in steps (4-9) and (4-10) is generally applicable to SLAM, no matter global or local. As previously discussed in Section 2.5.1, for real-time vehicle operations, it is only needed to maintain a local map that the vehicle has to deal with directly and immediately.

In local SLAM, we always maintain a local map for the surrounding environment of the vehicle. As presented in [Wang 2004] [Vu 2009], every time the vehicle arrives near the boundary of the current local map, a new local map is initialized; the pose of the new map is computed according to vehicle pose and the cells inside the intersection area of the two maps are copied from the old map to the new one.

A bit different from the practice in [Wang 2004] [Vu 2009], we do not store old local maps and only keeps the current local map, because we have no intention to carry out any task related to global mapping. Besides, in [Wang 2004] [Vu 2009], when the old local map is transformed into new local map, the map is not only translated but also

rotated so that the vehicle always starts at a fixed position with a fixed direction in the new local map. In our practice, we do not rotate the map and only translate the map by times of grid cell side length, for the sake of implementation convenience. We always keep the vehicle almost at the center of the local map. In other words, the local SLAM is more like sliding a local map window along the vehicle trajectory, as illustrated in Figure 4.4. An example of local occupancy grid map built during SLAM is illustrated in Figure 2.13.

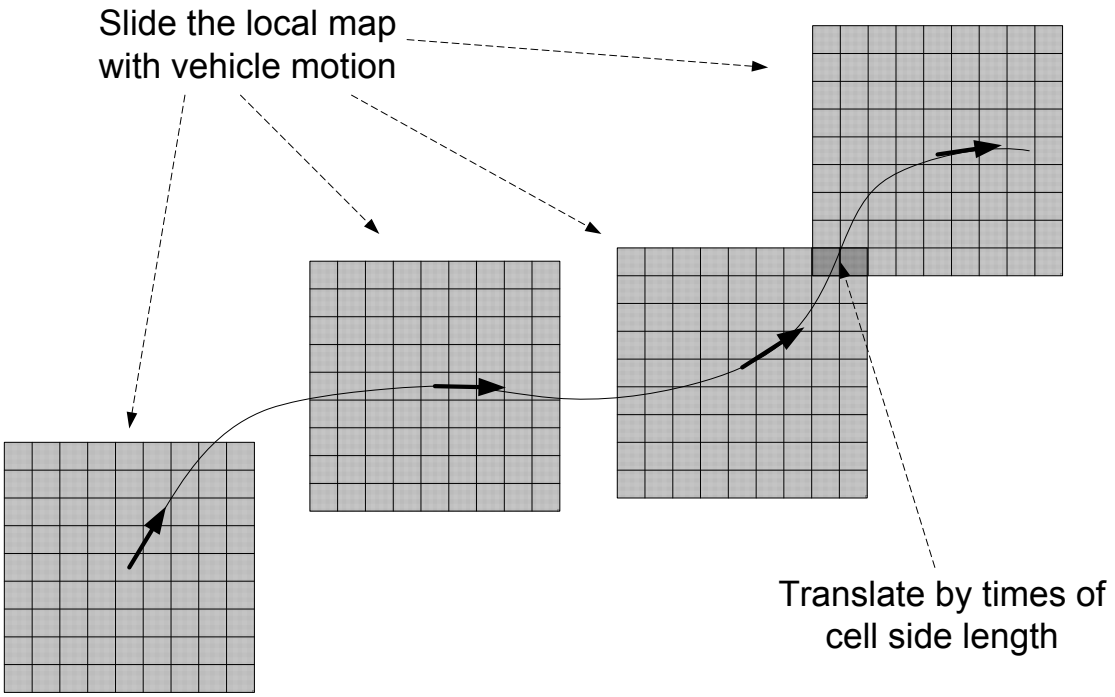


Figure 4.4 Slide the local occupancy grid map with vehicle motion

As briefly mentioned in Section 4.2.1, there are two sorts of vehicle state (pose): vehicle local state and vehicle global state. Vehicle local state is concerned in local SLAM as described in this section, whereas vehicle global state is concerned in vehicle global localization as described in Section 3.4.2 and Section 3.4.3.

The reason for not mixing vehicle global state and vehicle local state during local SLAM is that vehicle global state estimate might result in discontinuity of estimated vehicle trajectory and inconsistent mapping result. For example, imagine that the vehicle has no global (absolute) positioning measurement for a long duration and has a large deviation from its true position; at time *t*, the vehicle suddenly gets an ideally accurate global positioning measurement and update its state to the correct position.

Although the vehicle state estimate becomes accurate at time t, there turns to be a large discontinuity between the vehicle state estimate at time t-1 and that at time t, which will further result in spatially inconsistent mapping from time t-1 to time t.

In fact, the essential function of local SLAM is to establish consistent relative spatial relationship among the vehicle states and the surrounding environment at consecutive time sequence. Whether or not the vehicle is globally registered is not so important in local SLAM.

On the other hand, the vehicle local state estimate $\mathbf{S}_t$ in local SLAM can be used to correct original motion sensor measurements:

$$\Delta d_t^{\,c} = \sqrt{(\hat{x}_{s(t)} - \hat{x}_{s(t-1)})^2 + (\hat{y}_{s(t)} - \hat{y}_{s(t-1)})^2}$$
$$\Delta \theta_t^{\,c} = \hat{\theta}_{s(t)} - \hat{\theta}_{s(t-1)}$$

The corrected motion estimates Δd_t^c and $\Delta \theta_t^c$ are then used to evolve vehicle global state $\mathbf{X}_t$ for global localization.

For map representation convenience, the local map coordinates system is always attached to the local map. Let the local map coordinate system be denoted as $\mathbf{T}_t$; we always set $\mathbf{T}_0$ to be $\mathbf{0}$. The local map window is translated along the vehicle trajectory by times of grid cell side length, i.e.

$$\mathbf{T}_t = \mathbf{T}_{t-1} + \begin{bmatrix} n_{x(t)} \\ n_{y(t)} \\ 0 \end{bmatrix} l_c$$

The $n_{x(t)}$ and $n_{y(t)}$ are the number of cells by which the local map is translated respectively along the x direction and the y direction; the l_c is the grid cell side length. The vehicle local state $\mathbf{S}_t$ in the local map is also translated accordingly. In local SLAM, when we estimate $\mathbf{S}_t$ from $\mathbf{S}_{t-1}$ and $\mathbf{M}_{t-1}$, the $\mathbf{S}_t$ is actually represented in $\mathbf{T}_{t-1}$. As the local map is slid to $\mathbf{T}_t$, we have also to transform the $\mathbf{S}_t$ by subtracting from it the translation between $\mathbf{T}_{t-1}$ and $\mathbf{T}_t$, i.e.

$$\mathbf{S}_t = \mathbf{S}_t - (\mathbf{T}_t - \mathbf{T}_{t-1})$$

Given vehicle local state $\mathbf{S}_t$ and vehicle global state $\mathbf{X}_t$, for a point, the transformation between its coordinates in the local map ($\mathbf{p}_L$) and its coordinates in the global reference ($\mathbf{p}_G$) is as follows:

$$\mathbf{p}_G = \mathbf{X}_t \oplus inv(\mathbf{S}_t) \oplus \mathbf{p}_L$$
$$\mathbf{p}_L = \mathbf{S}_t \oplus inv(\mathbf{X}_t) \oplus \mathbf{p}_G$$

4.3 Occupancy Grid Maps Merging

In previous sub-section, we have described occupancy grid based local SLAM method for single vehicle operation. In this sub-section, we introduce a new method of occupancy grid maps merging, which are used to associate the perceptions of different intelligent vehicles.

4.3.1 Merging Framework

Given two vehicles A and B; following the denotations used previously, let their global states be denoted as $\mathbf{X}_{A(t)}$ and $\mathbf{X}_{B(t)}$, their local states be denoted as $\mathbf{S}_{A(t)}$ and $\mathbf{S}_{B(t)}$, their occupancy grid maps be denoted as $\mathbf{M}_{A(t)}$ and $\mathbf{M}_{B(t)}$, and their occupancy grid maps coordinates systems be denoted as $\mathbf{T}_{A(t)}$ and $\mathbf{T}_{B(t)}$. Without loss of generality, we neglect the subscript of time index t during following description. Then $\mathbf{M}_A$ and $\mathbf{M}_B$ are intended to be merged.

The process of occupancy grid maps merging can be generalized as the following optimization problem: First, design an objective function F_c in terms of two arbitrary occupancy grid maps $\mathbf{M}_1$ and $\mathbf{M}_2$, i.e. $F_c(\mathbf{M}_1, \mathbf{M}_2)$, which is used to measure their consistency degree. Second, search the optimal relative pose $\mathbf{p}_{BA}$ that maximizes the consistency measure between $\mathbf{M}_A$ and $\mathbf{p}_{BA} \oplus \mathbf{M}_B$, i.e.

$$\hat{\mathbf{p}}_{BA} = \arg\max_{\mathbf{p}_{BA}} F_c(\mathbf{M}_A, \mathbf{p}_{BA} \oplus \mathbf{M}_B) \tag{4-11}$$

In fact, the process of occupancy grid maps merging also includes a step of integrating the two occupancy grid maps into one map after they have been aligned correctly according to the estimated relative pose $\mathbf{p}_{BA}$. On the other hand, this last step would be comparatively trivial if the two occupancy grid maps can be aligned correctly. Besides, we may not need to use the integrated map; a more general and more fundamental role that the occupancy grid maps merging assumes is to spatially relate different intelligent

vehicles to each other, so that one vehicle can have meaningful inference about the perception of another vehicle. Therefore, when we mention occupancy grid maps merging, we do not necessarily indicate the practice of integrating two occupancy grid maps into one map; we would rather treat occupancy grid maps merging as a way to perform perceptions association.

4.3.2 The Objective Function based on Occupancy Likelihood

In [Birk & Carpin 2006], the objective function F_c consists of a *similarity* term and a *lock* term: the similarity term which is based on a *distance-map* represents the overall distances between the maps to-be-merged; the lock term is a part heuristically added to counteract the over-fitting effect.

$$F_c(\mathbf{M}_1, \mathbf{M}_2) = \sum_{c \in \{occ, free\}} \frac{\sum_{\mathbf{M}_1[m_1]=c} \min\{md(m_1, m_2) \mid \mathbf{M}_2[m_2] = c\}}{\#_c(\mathbf{M}_1)} + c_{lock} \cdot (\mathrm{dis}(\mathbf{M}_1, \mathbf{M}_2) - \mathrm{agr}(\mathbf{M}_1, \mathbf{M}_2)) \tag{4-12}$$

This objective function in [Birk & Carpin 2006] has two major disadvantages: first, the parameter c_{lock} in the heuristically added lock term has to be tuned empirically according to concrete scenarios. Second, this objective function is sensitive to maps inherent inconsistency i.e. maps inconsistency that still exists even if the maps to-be-merged are aligned correctly. Maps inherent inconsistency can be caused by dynamic entities which are common in outdoor environment. Maps inherent inconsistency can also be caused by the inconsistency of perception poses at different vehicles; for example, the same environment might appear noticeably different if it is scanned by laser scanners at different heights. For the objective function in [Birk & Carpin 2006], maps inherent inconsistency would cause drastic value change in the distance-map based similarity term and false counting of *agreement* and *disagreement* in the lock term.

Here, we propose an objective function based on occupancy likelihood, similar to the idea of the occupancy grid based scan matching as introduced in [Vu 2009] (see Section 4.2.3). Let the occupied cells with local maximum occupancy state (referred to as *local maximum occupied cells*) in $\mathbf{M}_A$ and $\mathbf{M}_B$ be respectively denoted as a set of two-dimensional points $\{\mathbf{o}_{A(1)}, \mathbf{o}_{A(2)}, \ldots, \mathbf{o}_{A(na)}\}$ and another set $\{\mathbf{o}_{B(1)}, \mathbf{o}_{B(2)}, \ldots, \mathbf{o}_{B(nb)}\}$—If the range scan has no measurement errors and is always ideally situated in the same scanning plane relative to the environment, then the occupancy grid map will be

comprised of a group of fine occupied border lines with only one cell width, which correspond to the intersection border lines between the scanning plane and the surfaces of environment objects. In reality, however, an occupancy grid map usually does not have such fine object border lines due to various sorts of errors. Therefore, we only select those local maximum occupied cells because they are most likely to be located on true objects borders—Let the occupancy state of a point **p** in an occupancy-grid map **M** be denoted as **M**(**p**); then the objective function F_c is defined as in (4-13):

$$\begin{aligned} F_c(\mathbf{M}_{\mathrm{A}}, \mathbf{p}_{\mathrm{BA}} \oplus \mathbf{M}_{\mathrm{B}}) = & \sum_{i=1}^{nb} \mathbf{p}_{\mathrm{BA}} \oplus \mathbf{o}_{\mathrm{B}(i)} \in Occ(\mathbf{M}_{\mathrm{A}}) \{\mathbf{M}_{\mathrm{A}}(\mathbf{p}_{\mathrm{BA}} \oplus \mathbf{o}_{\mathrm{B}(i)})\} \\ & + \sum_{i=1}^{na} inv(\mathbf{p}_{\mathrm{BA}}) \oplus \mathbf{o}_{\mathrm{A}(i)} \in Occ(\mathbf{M}_{\mathrm{B}}) \{\mathbf{M}_{\mathrm{B}}(inv(\mathbf{p}_{\mathrm{BA}}) \oplus \mathbf{o}_{\mathrm{A}(i)})\} \end{aligned} \tag{4-13}$$

Equation computes the occupancy likelihood sum of the local maximum occupied cells of $\mathbf{M}_{\mathrm{B}}$ in $\mathbf{M}_{\mathrm{A}}$ together with that of $\mathbf{M}_{\mathrm{A}}$ in $\mathbf{M}_{\mathrm{B}}$. The *Occ* means the set of occupied cells, which are selected by a threshold. Here, the *Occ* threshold is not intended to determine whether a grid cell is truly occupied or not in reality; it is only used to select grid cells that tend to be occupied or be closer to truly occupied cells. So there is fair flexibility in setting this threshold. We can set the threshold to be just above the unknown occupancy state (i.e. 0.5), because according to the inverse measurement model introduced in Section 4.2.2, a grid cell with occupancy state above 0.5 even is not truly occupied, yet is at least no far away from truly occupied cells. For example, we can set the *Occ* threshold to be 0.6.

In practice, a simplification of (4-13) , i.e. only computing the occupancy likelihood sum of the local maximum occupied cells of one map in the other without computing the converse part, would achieve fairly desirable performance as well, as in (4-14):

$$F_c(\mathbf{M}_{\mathrm{A}}, \mathbf{p}_{\mathrm{BA}} \oplus \mathbf{M}_{\mathrm{B}}) = \sum_{i=1}^{nb} \mathbf{p}_{\mathrm{BA}} \oplus \mathbf{o}_{\mathrm{B}(i)} \in Occ(\mathbf{M}_{\mathrm{A}}) \{\mathbf{M}_{\mathrm{A}}(\mathbf{p}_{\mathrm{BA}} \oplus \mathbf{o}_{\mathrm{B}(i)})\} \tag{4-14}$$

The occupancy likelihood based objective functions in (4-13) and (4-14) only take into account the consistent part of the occupancy grid maps to-be-merged. In other words, they are not influenced by maps inconsistent part. For example, if an area in one map tends to be occupied whereas its counterpart in the other map tends to be free, then this area is not taken into account in the proposed objective function and naturally will not

have influence on the output of the latter. Therefore, the proposed objective function is insensitive to maps inherent inconsistency.

For local maps of enough size, stable and consistent objects (buildings, infrastructures etc) are usually the dominating factors. These dominant consistent objects always contribute to successful local maps merging, because the correct maps relative pose which also aligns these consistent objects correctly normally achieves distinguished high value with the proposed objective function.

If we set the occupancy state space to be comprise of only three discrete states 0 (free), 0.5 (unknown), and 1 (occupied), then the occupancy likelihood based objective function is reduced to a simple formula similar to the *agr* term in the objective function in [Birk & Carpin 2006] which discretely computes the number of consistent cells, as follows:

$$F_c(\mathbf{M}_\mathrm{A}, \mathbf{p}_\mathrm{BA} \oplus \mathbf{M}_\mathrm{B}) \propto \mathrm{agr}(\mathbf{M}_1, \mathbf{M}_2) = \#\{\mathbf{p} \mid \mathbf{M}_1(\mathbf{p}) = \mathbf{M}_2(\mathbf{p}) \in C\}$$

However, we try to avoid any preliminary step of discretization on the occupancy state space that is originally continuous, because a discretization step is unnecessary and it reduces the information originally contained in the occupancy grid map. Besides, the practice of discretization, especially the triplet discretization, can worsen the optimization structure and make it more difficult to search the correct solution. Therefore, we prefer the occupancy likelihood based objective function as in (4-13) or (4-14) that is directly applicable to merging general occupancy grid maps whose state spaces are continuous.

The *distance-map* based *similarity* term in (4-12) [Birk & Carpin 2006] might smoother the value space of the objective function; an idea is to incorporate this similarity term into (4-14) (or (4-13)) and we have (4-15):

$$\begin{aligned} F_c(\mathbf{M}_\mathrm{A}, \mathbf{p}_\mathrm{BA} \oplus \mathbf{M}_\mathrm{B}) = & \frac{1}{nb}\sum_{i=1}^{nb} \min\{md(\mathbf{o}_\mathrm{A}, \mathbf{p}_\mathrm{BA} \oplus \mathbf{o}_{\mathrm{B}(i)}) \mid \mathbf{o}_\mathrm{A} \in Occ(\mathbf{M}_\mathrm{A})\} \\ & + c_{lock} \cdot \frac{1}{nb}\sum_{i=1}^{nb} \mathbf{p}_\mathrm{BA} \oplus \mathbf{o}_{\mathrm{B}(i)} \in Occ(\mathbf{M}_\mathrm{A}) \{\mathbf{M}_\mathrm{A}(\mathbf{p}_\mathrm{BA} \oplus \mathbf{o}_{\mathrm{B}(i)})\} \end{aligned} \tag{4-15}$$

As will be demonstrated in experiments in Section 6.4.5, a large c_{lock} in (4-12) or (4-15) would make the objective function comparatively robust to maps inherent inconsistency and always enables finding the correct alignment (if suitable optimization technique is used, as will be introduced in the next sub-section). With a large c_{lock}, the distance-map

based similarity term has trivial contribution and can be totally neglected for simplicity. This is the reason why we propose to use the objective function (4-13) (or (4-14) for simplicity).

4.3.3 Optimization using Genetic Algorithm

The initial value of $\mathbf{p}_{BA}$ can be computed with GPS based vehicle global localization results, yet this initial value might be far away from the global optimal maps alignment. For intelligent vehicle systems with low-accuracy GPS, the initial position error of $\mathbf{p}_{BA}$ can be twenty meters; initial orientation error of $\mathbf{p}_{BA}$ can also be large. Besides, the value space of the objective function is normally multimodal and of irregular shape on the whole. Therefore, local optimization searching techniques such as gradient based analytical techniques tend to fail when facing such large initial estimate error.

The strategy of evolutionary genetic algorithm [Man *et al.* 1999] is adopted to solve the optimization problem (4-11). One important motivation for using genetic algorithm is that it is independent of the objective function value space and it is ready to solve multimodal, non-differentiable, or non-continuous problems. Another motivation lies in its intrinsic parallelism architecture, which makes it directly suitable for parallel computation framework if needed. Besides, it can be well implemented in a dynamic (or recursive) scheme for real-time vehicle operation.

Genetic algorithm is rather a methodology instead of being a list of concrete execution procedures. As an analogy to species evolution under the influence of natural selection, the fundamental spirit of genetic algorithm is to evaluate the fitness values of a group of tentative solution individuals, vary them with biologically inspired operations such as crossover and mutation, and keep those better individuals. The concrete procedures to put this spirit into practice are problem oriented and can be specially designed and modified. The concrete procedures in our implementation are as follows:

1. *Initialization*: randomly generate an initial population of $\mathbf{p}_{BA}$:

(1-a) Compute the initial value of $\mathbf{p}_{BA}$ with GPS based global localization results of the two vehicles:

$$\mathbf{p}_{BA(init)} = \mathbf{S}_A \oplus inv(\mathbf{X}_A) \oplus \mathbf{X}_B \oplus inv(\mathbf{S}_B)$$

(1-b) In a certain error range around $\mathbf{p}_{BA(init)}$, randomly generate an initial population of $\mathbf{p}_{BA}$ i.e. $\{\mathbf{p}_{BA(k)} | k=1,2,\ldots,n\}$. With an intention to examine the robustness of the method, we deliberately exaggerate this initial error range to be $\pm$30 meters in position and $\pm$30

degrees in orientation. The initial error range can also be estimated from the covariance estimates of the vehicles global states; a better estimation of the initial error range might accelerate the optimization searching process.
Concerning the representation of a generic individual $\mathbf{p}_{\mathrm{BA}(k)}$ in the population, we do not make *bit (binary) string encoding* [Man *et al.* 1999] on $\mathbf{p}_{\mathrm{BA}(k)}$ as originally in genetic algorithm; instead, we directly handle the real-value vector form of $\mathbf{p}_{\mathrm{BA}(k)}$ for implementation convenience.

2. *Evolution*: iteratively perform the following sub-steps as follows:
(2-a) Compute the likelihood value (or *fitness value* in traditional genetic algorithm terms) of each individual in the population, according to (4-14).
(2-b) Compute mean likelihood value of the population. For an individual, if its likelihood value is above the mean likelihood value, assign the individual to the *elite group*; otherwise, assign it to the *inferior group*. $\{\mathbf{p}_{\mathrm{BA}(k)} | k=1,2,\ldots,n\}$

$$F_{c(\mathrm{mean})} = \frac{1}{n}\sum_{k=1}^{n} F_c(\mathbf{M}_{\mathrm{A}}, \mathbf{p}_{\mathrm{BA}(k)} \oplus \mathbf{M}_{\mathrm{B}})$$

$$\{\mathbf{p}_{\mathrm{BA(elite)}}\} = \{\mathbf{p}_{\mathrm{BA}(i)} \mid F_c(\mathbf{M}_{\mathrm{A}}, \mathbf{p}_{\mathrm{BA}(i)} \oplus \mathbf{M}_{\mathrm{B}}) \geq F_{c(\mathrm{mean})}\}$$

$$\{\mathbf{p}_{\mathrm{BA(inferior)}}\} = \{\mathbf{p}_{\mathrm{BA}(j)} \mid F_c(\mathbf{M}_{\mathrm{A}}, \mathbf{p}_{\mathrm{BA}(j)} \oplus \mathbf{M}_{\mathrm{B}}) < F_{c(\mathrm{mean})}\}$$

Categorizing the population according to their mean likelihood value is a simple yet effective way to decide which individuals are more likely to survive and more likely to have influence on the following generation.
(2-c) Mutate the individuals in the elite group. For an individual, if its mutation has higher likelihood value than its own, then replace this individual with its mutation; otherwise, just keep this individual originally in the elite group.

$$FOR\ \mathbf{p}_{\mathrm{BA}(k)} \in \{\mathbf{p}_{\mathrm{BA(elite)}}\}$$
$$\mathbf{p}_{\mathrm{BA}(k)}{}^{*} = \mathbf{mutate}(\mathbf{p}_{\mathrm{BA}(k)})$$
$$IF\ F_c(\mathbf{p}_{\mathrm{BA}(k)}{}^{*}) > F_c(\mathbf{p}_{\mathrm{BA}(k)})\ THEN\ \mathbf{p}_{\mathrm{BA}(k)} = \mathbf{p}_{\mathrm{BA}(k)}{}^{*}$$
$$END\ FOR$$

Among the elite group, the best individual is an exception, which gets more times (for example, 100 times) of mutation. If no mutation is better, then just keep the best individual unchanged; otherwise, keep the best mutation to replace the original best

individual. (**Note**: this is like a coarse-to-fine strategy executed for the best individual, which is intended to refine the precision of the best individual of maps alignment. On the other hand, a simple coarse-to-fine strategy for the overall optimization process tends to fail, because of the sparsity of the individuals and the irregularity of the value space; the best individual in one evolution does not guarantee reducing the error range largely)

(2-d) Replace the inferior group with new individuals; more specifically, replace each individual in the inferior group with a new individual that is generated from old individuals by applying the following genetic operations with specified probabilities:

(2-d-i) Copy the best individual (only performed once).

(2-d-ii) Randomly select an individual from the elite group and mutate it to be the new individual.

(2-d-iii) Randomly select two individuals from the elite group, create a new individual by executing *crossover* on them and mutating the crossover result. Two sorts of crossover are designed:

Crossover I: Mix the position parts and orientation parts of the two individuals. Let the two elite individuals be denoted as $\mathbf{p}_{\mathrm{BA(e1)}}=[x_{\mathrm{BA(e1)}}, y_{\mathrm{BA(e1)}}, \theta_{\mathrm{BA(e1)}}]^{\mathrm{T}}$ and $\mathbf{p}_{\mathrm{BA(e2)}}=[x_{\mathrm{BA(e2)}}, y_{\mathrm{BA(e2)}}, \theta_{\mathrm{BA(e2)}}]^{\mathrm{T}}$; the new individual is generated as follows:

$$\begin{aligned}\mathbf{p}_{\mathrm{BA(new)}} &= \mathbf{crossover\_I}(\mathbf{p}_{\mathrm{BA(e1)}}, \mathbf{p}_{\mathrm{BA(e2)}}) \\ &= [x_{\mathrm{BA(e1)}}, y_{\mathrm{BA(e1)}}, \theta_{\mathrm{BA(e2)}}]^{T} \text{ or } [x_{\mathrm{BA(e2)}}, y_{\mathrm{BA(e2)}}, \theta_{\mathrm{BA(e1)}}]^{T}\end{aligned}$$

Crossover II: Make a random linear combination of the two individuals (the λ is a randomly generated real value in [0, 1]:

$$\mathbf{p}_{\mathrm{BA(new)}} = \mathbf{crossover\_II}(\mathbf{p}_{\mathrm{BA(e1)}}, \mathbf{p}_{\mathrm{BA(e2)}}) = \lambda \mathbf{p}_{\mathrm{BA(e1)}} + (1-\lambda)\mathbf{p}_{\mathrm{BA(e2)}}$$

The *Crossover I* and *Crossover II* are specially designed crossover operations to directly hand the real-value vector form of $\mathbf{p}_{\mathrm{BA}(k)}$, yet we can make an analogy between them and the traditional bit string based crossover operations by adopting the *Building Block Hypothesis* [Man *et al.* 1999]. For a generic individual $\mathbf{p}_{\mathrm{BA}(k)}$, if we believe that its position component and orientation component are basic building blocks which contribute to its fitness, then the traditional crossover operation turns to be *Crossover I*, as demonstrated by Figure 4.5(top). If we believe that the proportion between its position component and orientation component is basic building block which

contributes to its fitness, then the traditional crossover operation turns to be *Crossover II*, as demonstrated by Figure 4.5(bottom).

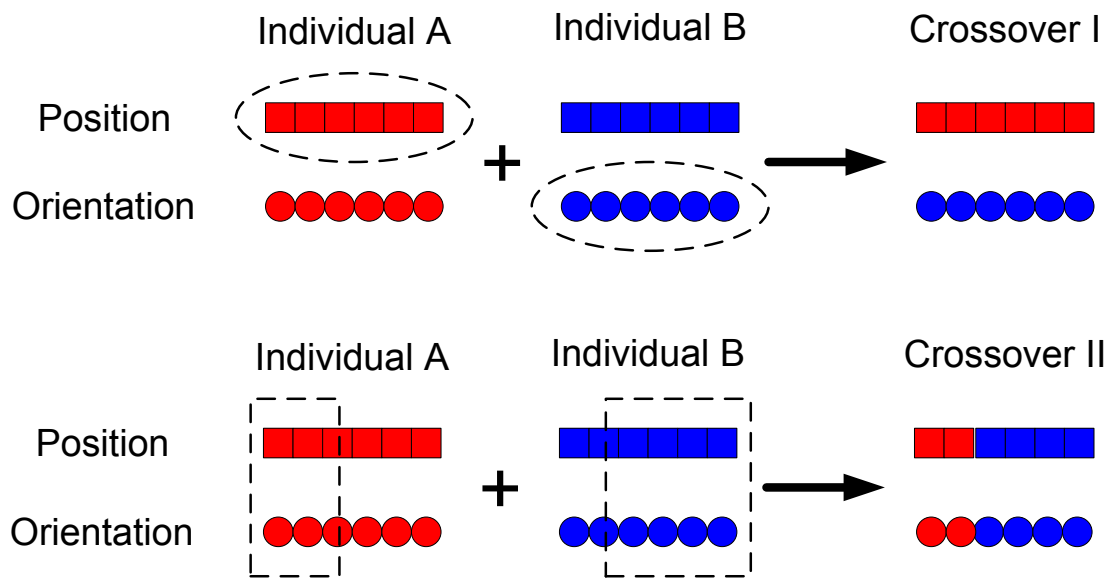


Figure 4.5 Crossover operation: (top) crossover I; (bottom) crossover II

It is worthy noting that the *Building Block Hypothesis* is a theory which does not strictly prove but heuristically explain the working mechanism of genetic algorithm. The *Building Block Hypothesis* can be rather treated as a guide for designing the genetic operations.

(2-d-iv) *Re-initialization*: Create the new individual according to GPS based vehicle global localization results and the error range, as in the initialization process. This re-initialization practice is to keep the diversity of the population.

When two vehicles meet or re-meet, the *initialization* step is performed once and the sub-steps in *evolution* are repeatedly performed. A dynamic scheme of the genetic algorithm is used: the generation of $\mathbf{p}_{BA}$ individuals from last period is propagated to the current period, according to the change of local map coordinates systems, i.e. (for k=1,2,…,n):

$$\mathbf{p}_{BA(k)(t)} = [inv(\mathbf{T}_{A(t)}) \oplus \mathbf{T}_{A(t-1)}] \oplus \mathbf{p}_{BA(k)(t-1)} \oplus [inv(\mathbf{T}_{B(t-1)}) \oplus \mathbf{T}_{B(t)}]$$

As long as the vehicles are in the neighborhood and in cooperation, the *evolution* step can be performed unceasingly. As a result, we only need to assign few times of

evolution for each period (for example, once), which largely reduces computational burden at one period; moreover, as the evolution continuous unceasingly, the dynamic scheme of genetic algorithm will finally converge to the optimum. In our tests, the genetic algorithm usually converges to the optimum in only few periods (no more than one second).

An example of the local occupancy grid maps merging result is illustrated in Figure 4.6.

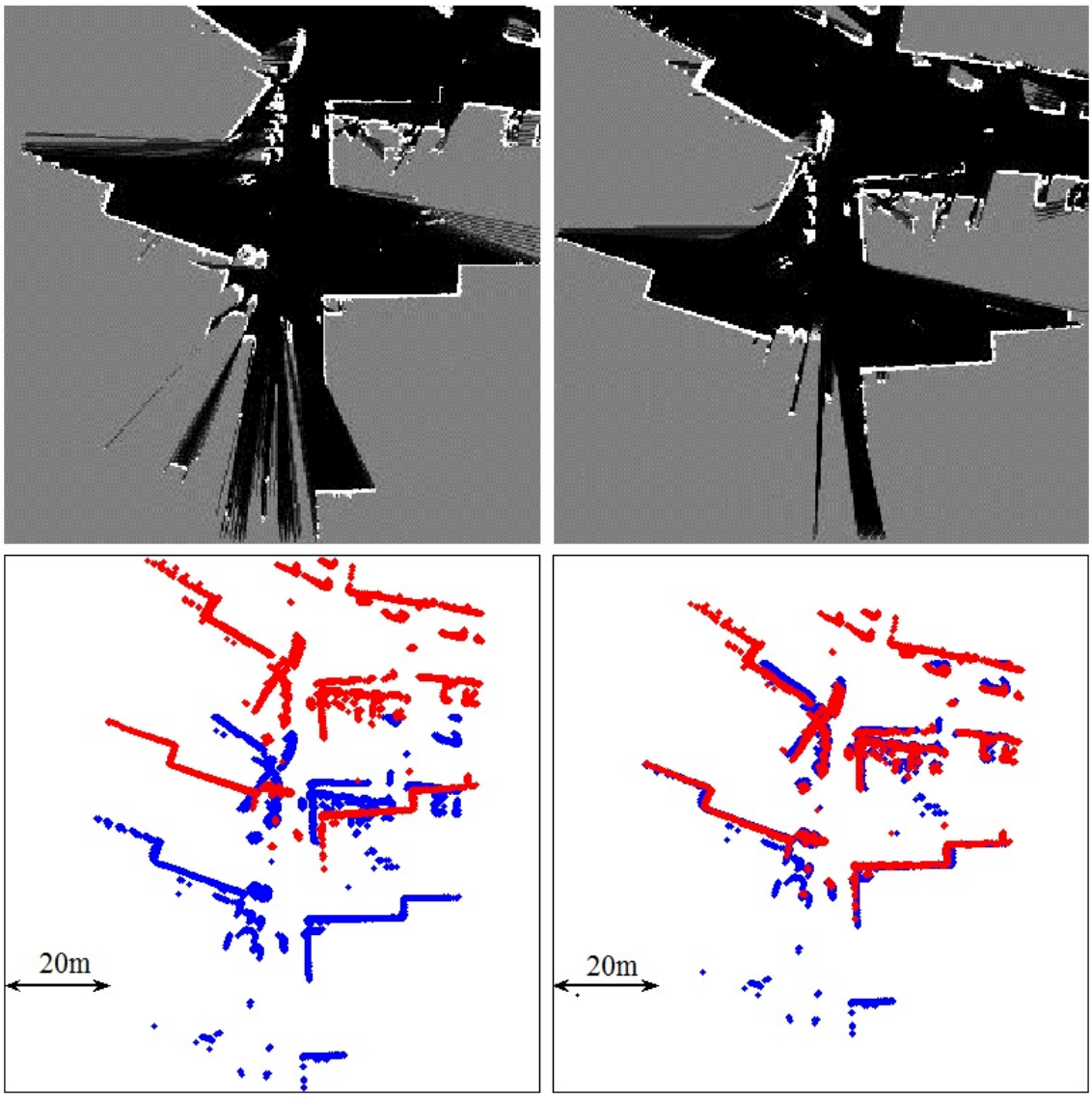


Figure 4.6 (top) local SLAM results; (bottom-left) maps alignment according to low-accuracy GPS based localization results; (bottom-right) maps merging result

4.4 Cooperative Moving Objects Detection

The purpose of this sub-section is not to propose certain moving object detection method, but to demonstrate a scheme of multi-vehicles cooperative moving objects detection: the occupancy grid maps merging method is used to associate the perceptions of different vehicles; based on the perceptions association result, the moving objects detected by a vehicle can be mapped into the local map of another vehicle and merged with the moving objects detected by the latter.

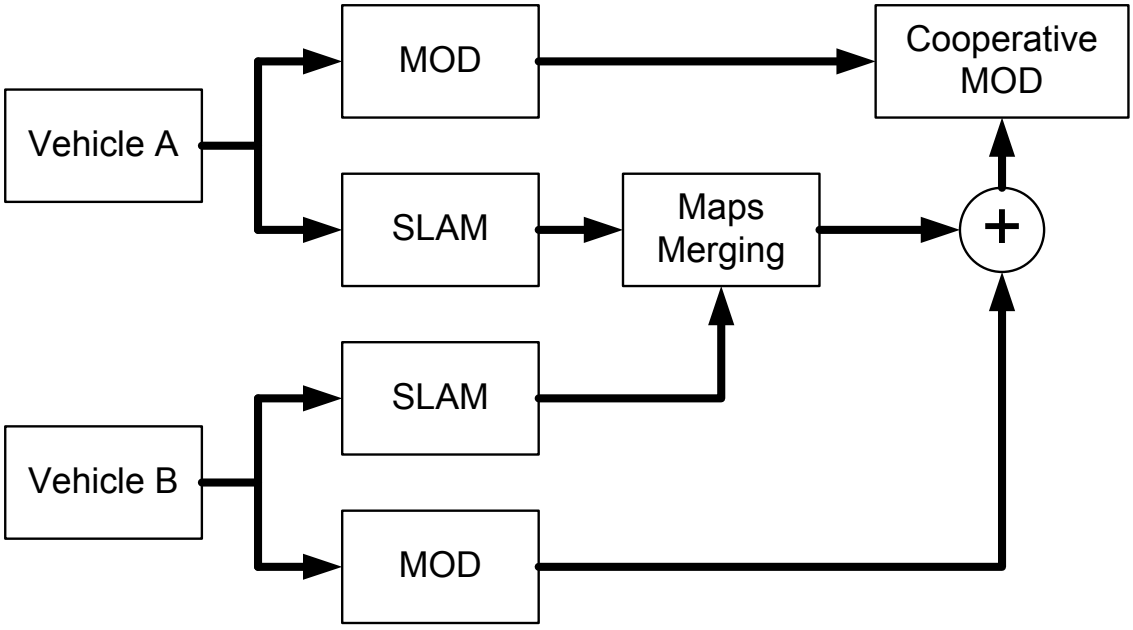


Figure 4.7 Cooperative moving objects detection (Cooperative MOD)

To demonstrate a complete application of cooperative moving objects detection, we incorporate two basic moving objects detection methods, namely *consistency-based detection* and *moving object map based detection* [Wang 2004] [Vu 2009]. More sophisticated moving objects detection methods can also be incorporated into the local occupancy grid maps merging based scheme of cooperative local mapping and moving objects detection.

Consistency-based detection: given a new scan of range measurements and previously constructed occupancy grid maps, the idea is to find the inconsistent part between range measurements and free space in the local occupancy grid map. If a range point is detected on a location of previously free space, then it is regarded as a moving point.

The range data are clustered into segments; for a segment, if the number of moving points is larger than a half of the total points, then the segment is identified as potential moving object.

Moving object map based detection: a local moving object map is created to store information about previously detected moving objects; each cell in the moving object map stores a value indicating the number of observations that a moving object has been observed at that cell location. If the cell value is above certain threshold, the range point associated with this cell is regarded as a moving point.

During multi-vehicles cooperation, a vehicle (referred to as *ego vehicle*) will merge the local occupancy grid map of another vehicle into its own occupancy grid map, using introduced occupancy grid maps merging method. The detected moving objects of another vehicle can also be transformed into the ego vehicle reference and fused with the detected moving objects of the ego vehicle: if a detected moving object of another vehicle and a detected moving object of the ego vehicle have at least partial overlap, then the two objects are regarded as the same object and fused into one object.
For the ego vehicle, the merged occupancy grid map which incorporates the data of another vehicle is only used for current time and will not be used during the following SLAM. In other words, the ego vehicle performs SLAM only based on its own sensor data. This is for guaranteeing the independence among the occupancy grid maps estimated by different vehicles. (**Note**: as we adopt the incremental maximum likelihood estimation framework to handle local SLAM with large size occupancy grid map, we do not maintain posterior estimate uncertainty; therefore, the strategy of applying the split covariance intersection filter as in CHAPTER 3 is not applicable here, not to mention the large computational burden if applicable. On the other hand, in future, it deserves finding a solution to enable a vehicle to utilize the merged map during its following process of SLAM and MOD. This future extension will be discussed in details in the last chapter of this dissertation)

4.5 Summary

We have reviewed the method of occupancy grid based single vehicle local SLAM, including how to use laser scanner based range measurements to incrementally update the occupancy grid map estimate according to the inverse measurement model and how to estimate current vehicle local state (pose) with last estimate of vehicle local state and occupancy grid map. We have explained the different roles of vehicle local state and

vehicle global state; we have described how vehicle local state estimate in SLAM can be used to assist vehicle global state estimation. We have presented the framework for occupancy grid maps merging by generalizing its essential part into an optimization problem; we have proposed a new objective function that measures the consistency degree of maps alignment based on occupancy likelihood. We have adopted the spirit of genetic algorithm and designed a set of concrete procedures to search the optimal maps alignment. We have introduced the scheme of multi-vehicles cooperative moving objects detection based on occupancy grid maps merging; for a complete implementation, we have reviewed two basic moving objects detection methods, namely the consistency-based detection and the moving object map based detection.

CHAPTER 5 Cooperative Augmented Reality

Résumé

Nous avons exploité la notion de réalité augmentée à la perception coopérative, formalisant ainsi le concept de *« réalité augmentée coopérative »* appliquée au contexte des systèmes de véhicules intelligents. Nous nous sommes intéressés particulièrement au scénario de véhicules « leader-suiveur » auquel l'approche de réalité augmentée est appliquée. Pour cela, nous utilisons deux capteurs : un télémètre laser et une caméra. Nous avons décrit comment établir une relation spatiale entre deux vues selon la géométrie perspective. Nous avons introduit une technique permettant à un télémètre laser 2D de fournir à une caméra des données lui permettant d'estimer la profondeur de perception visuelle. Nous avons présenté la façon de projeter la perception d'un véhicule sur celle d'un autre véhicule, en respectant la géométrie perspective décrite. Nous avons également introduit une nouvelle méthode de calibration extrinsèque pour une caméra et un télémètre laser 2D.

5.1 Introduction

In this chapter, we introduce a brand new idea of cooperative augmented reality which utilizes the results of cooperative local mapping to realize certain augmented reality effect. More specifically, we realize an augmented effect of 'seeing' through front vehicle, based on the intelligent vehicle sensor configurations described in Section 2.2. First, we describe the mathematical foundations of pinhole camera model and multi-views perspective geometry. Then, we introduce a technique of utilizing a 2D laser scanner to assist a mono-camera in estimating the visual perception depth approximately. Next, we present the process of realizing perspective transformation between the visual perceptions of two vehicles. As part of the provided solution, a new method is proposed for extrinsic co-calibration of a camera and a 2D laser scanner; the proposed calibration method reveals all the spatial relationships among the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system.

5.2 Front-Following Vehicles Scenario

Given a scenario of two vehicles: the front (first) vehicle and the following (second) vehicle; the front vehicle occludes the view of the following vehicle, as illustrated in Figure 5.1. This front-following vehicles scenario is typical in traffic environment and is potential dangerous, especially in some occasions such as the overtaking occasion (see Figure 1.7).

For this front-following vehicles scenario, the idea of cooperative augmented reality is to project the visual perception of the front vehicle onto that of the following vehicle, abiding by perspective geometry. In other words, we patch the occluded part of the view of the following vehicle with corresponding part of the view of the front vehicle. This is not simply a process of partial view copying and pasting between the two vehicles; we have to transform the partial view of the front vehicle according to perspective geometry, in order to make a vivid and natural reproduction of this partial view for the following vehicle, as if the following vehicle can directly see into the occluded area, as demonstrated in Figure 2.15.

Figure 5.1 Front-following vehicles scenario: (left) the view of the front vehicle; (right) the view of the following vehicle (occluded by the front vehicle)

5.3 Camera Model and Multi-Views Perspective Geometry

5.3.1 Pinhole Camera Model

Camera model is used to characterize the spatial (geometric) relationships between the objects perceived by the camera and their projections on the captured image. After removal of image distortion (or image distortion is negligible), the ideal pinhole camera model [Faugeras 1993] [Zhang 2000] can be adopted, as illustrated in Figure 5.2.

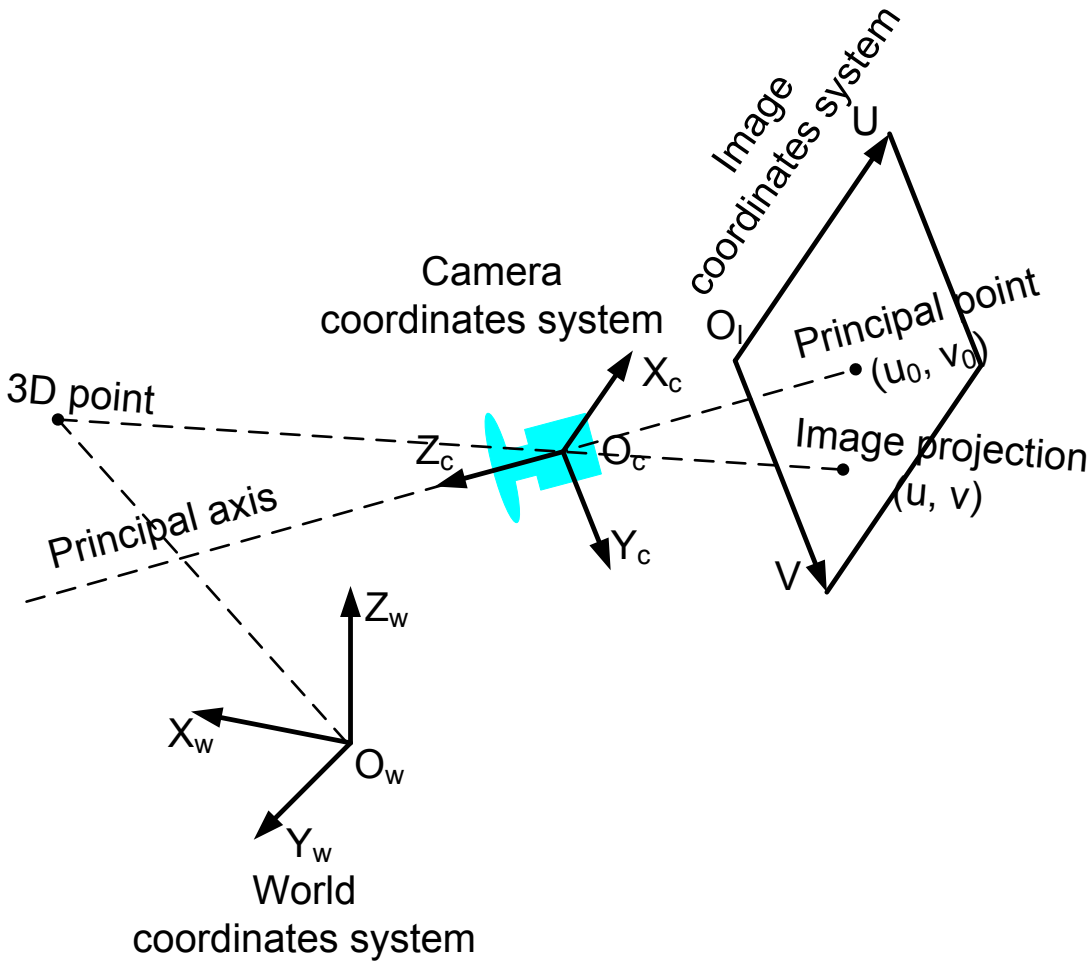


Figure 5.2 Pinhole camera model

There are three coordinates systems concerned in this model, namely the world coordinates system, the camera coordinates system, and the image coordinates system. The world coordinates system $\{\mathbf{O_w},\mathbf{X_w},\mathbf{Y_w},\mathbf{Z_w}\}$ is used to specify the location of environment objects in our 3-Dimensional (3D) world; it is usually established in *ad hoc* way which facilitates the fulfilling of certain tasks. The camera coordinates system $\{\mathbf{O_c},\mathbf{X_c},\mathbf{Y_c},\mathbf{Z_c}\}$ is attached to the camera, which is used to specify the location of environment objects relative to the camera; the origin $\mathbf{O_c}$ is situated on the virtual pinhole by which each projection line passes; the $\mathbf{Z_c}$ is aligned with the *principal axis* which is perpendicular to the image plane. The image coordinates system $\{\mathbf{O_i},\mathbf{U},\mathbf{V}\}$ is attached with the virtual 2D digital image plane in pixels (**Note**: in a camera, there is a physical image plane which captures environment light; digital image data are then

generated by digitizing the image (light) captured by the physical image plane. We can fairly imagine there is a virtual digital image plane overlapped with the physical image plane; the difference between them only lies in a conversion between physical unit and pixel unit).
The transformation from the world coordinates system to the camera coordinates system is given by a rotation $\mathbf{R_{wc}}$ and a translation $\mathbf{T_{wc}}$:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R_{wc}} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{T_{wc}} \tag{5-1}$$

The transformation from the camera coordinates system to the image coordinates system is given as follows:

$$\begin{aligned} u &= \alpha \frac{x_c}{z_c} + \gamma \frac{y_c}{z_c} + u_0 \\ v &= \beta \frac{y_c}{z_c} + v_0 \end{aligned} \tag{5-2}$$

The (u_0,v_0) is the location of the *principal point* (the intersection point of the principal axis and the image plane); the α and β are *scaling factors* in image u and v axes; the γ describes the skewness of two image axes. We can represent (5-2) in matrix form:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \quad \text{where} \quad \mathbf{A} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5-3}$$

The **A** is *camera intrinsic matrix*; the parameters in this matrix are *camera intrinsic parameters*, which can be calibrated using the chessboard plane based method [Zhang 2000]. In many applications, we can fairly assume that the scaling factors along two image axes are the same and the two image axes are strictly orthogonal, in other words, we assume that the image plane is ideally fabricated, then (5-2) and (5-3) turn into a simplified (yet effective) version, as in (5-4).

$$u = \alpha\frac{x_c}{z_c} + u_0 \qquad v = \alpha\frac{y_c}{z_c} + v_0 \quad \text{or} \quad \lambda\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & \alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \tag{5-4}$$

Combining (5-1) and (5-3), we obtain the transformation from the world coordinates system to the image coordinates system, as in (5-5).

$$\lambda\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{M}\begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad \text{where} \quad \mathbf{M} = \mathbf{A}\begin{bmatrix}\mathbf{R_{wc}} & \mathbf{T_{wc}}\end{bmatrix} \tag{5-5}$$

The **M** is the *perspective matrix*. The transformation described by (5-5) is referred to as *perspective mapping*. For a matrix **M**, we denote its *k*-th column vector as $\mathbf{M}_{(k)}$; for example, the perspective matrix **M** (always having four columns) can be represented in decomposed form as: $\mathbf{M}=[\mathbf{M}_{(1)}\ \mathbf{M}_{(2)}\ \mathbf{M}_{(3)}\ \mathbf{M}_{(4)}]$.

5.3.2 Multi-Views Perspective Geometry

Given two cameras (as illustrated in Figure 5.3) whose intrinsic matrixes are respectively denoted as $\mathbf{A_1}$ and $\mathbf{A_2}$ and whose spatial relationships with the world are respectively denoted as {$\mathbf{R_{wc1}}$ and $\mathbf{T_{wc1}}$} and {$\mathbf{R_{wc2}}$ and $\mathbf{T_{wc2}}$}; Let their perspective matrix be denoted as $\mathbf{M_1}$ and $\mathbf{M_2}$:

$$\mathbf{M}_1 = [\mathbf{M}_{1(1)}\quad \mathbf{M}_{1(2)}\quad \mathbf{M}_{1(3)}\quad \mathbf{M}_{1(4)}] = \mathbf{A}_1[\mathbf{R}_{\mathbf{wc}1}\quad \mathbf{T}_{\mathbf{wc}1}]$$
$$\mathbf{M}_2 = [\mathbf{M}_{2(1)}\quad \mathbf{M}_{2(2)}\quad \mathbf{M}_{2(3)}\quad \mathbf{M}_{2(4)}] = \mathbf{A}_2[\mathbf{R}_{\mathbf{wc}2}\quad \mathbf{T}_{\mathbf{wc}2}]$$

Let $\mathbf{P_w}=[x_w,y_w,z_w]^T$ denotes a 3D point in the world coordinates system; $\mathbf{P_{i1}}=[u_1,v_1]^T$ and $\mathbf{P_{i2}}=[u_2,v_2]^T$ denote the image points respectively in the first image coordinates system and the second image coordinates system. For an arbitrary vector **V**, let $\mathbf{V_{(A)}}$ denotes its augmented vector i.e. $\mathbf{V_{(A)}}=[\mathbf{V};1]^T$. Let $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ respectively denote the constant vectors $[1,0,0]^T$, $[0,1,0]^T$ and $[0,0,1]^T$.

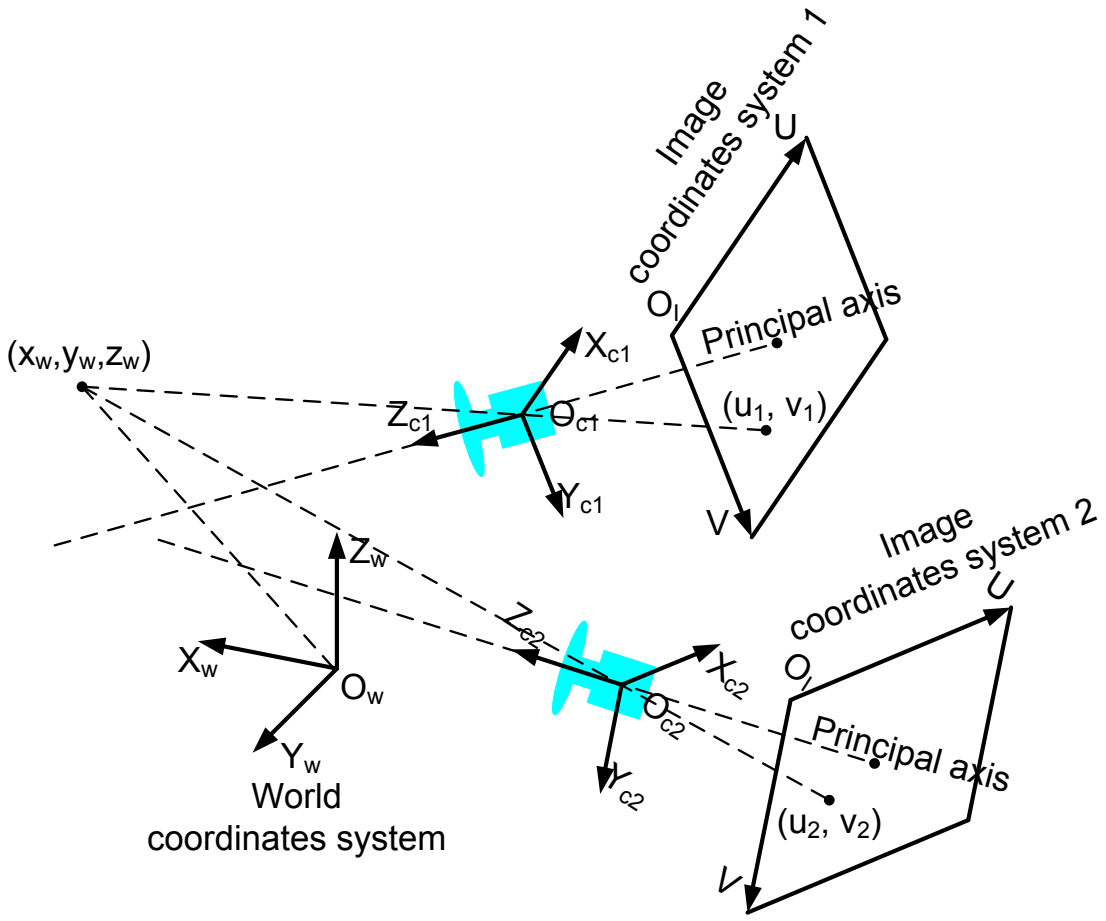


Figure 5.3 Multi-views perspective geometry

Formula I:
Consider the perspective mapping as specified in (5-5); given a known z_w, then x_w can be computed as:

$$x_w = \frac{\mathbf{e}_1^{\mathrm{T}}[\mathbf{M}_{(1)} \quad \mathbf{M}_{(2)} \quad z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}]^{-1}\mathbf{P_i}}{\mathbf{e}_3^{\mathrm{T}}[\mathbf{M}_{(1)} \quad \mathbf{M}_{(2)} \quad z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}]^{-1}\mathbf{P_i}} \tag{5-6}$$

Proof: With a known z_w, from (5-5) we have:

$$\lambda \mathbf{P_i} = \mathbf{M} \cdot \mathbf{P_w}$$
$$= x\mathbf{M}_{(1)} + y\mathbf{M}_{(2)} + z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}$$
$$= [\mathbf{M}_{(1)} \quad \mathbf{M}_{(2)} \quad z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}] \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix}$$

Then

$$\begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} = \lambda [\mathbf{M}_{(1)} \quad \mathbf{M}_{(2)} \quad z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}]^{-1} \mathbf{P_i}$$

Divide the first element by the third element in above right-side expression and then we have (5-6). End. □

Formula II:
Given the perspective mappings for two cameras:

$$\lambda_1 \mathbf{P_{i1(A)}} = \mathbf{M}_1 \cdot \mathbf{P_{w(A)}}$$
$$\lambda_2 \mathbf{P_{i2(A)}} = \mathbf{M}_2 \cdot \mathbf{P_{w(A)}}$$

For a 3D point $\mathbf{P_w}$, if its x_w coordinate is known, the geometric relationship between its projections on the two cameras image planes is given by (λ_{12} and λ_{21} are normalization constants):

$$\lambda_{12} \mathbf{P_{i1(A)}} = [\mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)} \ \ x_w\mathbf{M}_{1(1)} + \mathbf{M}_{1(4)}][\mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)} \ \ x_w\mathbf{M}_{2(1)} + \mathbf{M}_{2(4)}]^{-1} \mathbf{P_{i2(A)}}$$
$$\lambda_{21} \mathbf{P_{i2(A)}} = [\mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)} \ \ x_w\mathbf{M}_{2(1)} + \mathbf{M}_{2(4)}][\mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)} \ \ x_w\mathbf{M}_{1(1)} + \mathbf{M}_{1(4)}]^{-1} \mathbf{P_{i1(A)}} \quad (5\text{-}7)$$

Proof: With a known x_w we have:

$$\lambda_1 \mathbf{P_{i1(A)}} = \mathbf{M}_1 \cdot \mathbf{P_{w(A)}} = x_w\mathbf{M}_{1(1)} + y_w\mathbf{M}_{1(2)} + z_w\mathbf{M}_{1(3)} + \mathbf{M}_{1(4)}$$
$$= [\mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)} \ \ x_w\mathbf{M}_{1(1)} + \mathbf{M}_{1(4)}] \begin{bmatrix} y_w \\ z_w \\ 1 \end{bmatrix}$$

$$\lambda_2 \mathbf{P}_{\mathbf{i}2(\mathbf{A})} = \mathbf{M}_2 \cdot \mathbf{P}_{\mathbf{w}(\mathbf{A})} = x_w \mathbf{M}_{2(1)} + y_w \mathbf{M}_{2(2)} + z_w \mathbf{M}_{2(3)} + \mathbf{M}_{2(4)}$$

$$= [\mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)} \ \ x_w \mathbf{M}_{2(1)} + \mathbf{M}_{2(4)}] \begin{bmatrix} y_w \\ z_w \\ 1 \end{bmatrix}$$

Then

$$\lambda_1 [\mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)} \ \ x_w \mathbf{M}_{1(1)} + \mathbf{M}_{1(4)}]^{-1} \mathbf{P}_{\mathbf{i}1(\mathbf{A})}$$

$$= \begin{bmatrix} y_w \\ z_w \\ 1 \end{bmatrix} = \lambda_2 [\mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)} \ \ x_w \mathbf{M}_{2(1)} + \mathbf{M}_{2(4)}]^{-1} \mathbf{P}_{\mathbf{i}2(\mathbf{A})}$$

By rearranging the terms in above equation and we can get (5-7). End. □

Formula III:

For a 3D point $\mathbf{P}_\mathbf{w}$, if $\|\mathbf{P}_\mathbf{w}\| \rightarrow \infty$, the geometric relationship between its projections on the two cameras image planes is given by:

$$\begin{aligned} \lambda_{12} \mathbf{P}_{\mathbf{i}1(\mathbf{A})} &= [\mathbf{M}_{1(1)} \ \mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)}][\mathbf{M}_{2(1)} \ \mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)}]^{-1} \mathbf{P}_{\mathbf{i}2(\mathbf{A})} \\ \lambda_{21} \mathbf{P}_{\mathbf{i}2(\mathbf{A})} &= [\mathbf{M}_{2(1)} \ \mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)}][\mathbf{M}_{1(1)} \ \mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)}]^{-1} \mathbf{P}_{\mathbf{i}1(\mathbf{A})} \end{aligned} \tag{5-8}$$

Proof: We can verify that the following equation asymptotically holds as x_w increases to infinite:

$$\lim_{x_w \to \infty} [\mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)} \ \ x_w \mathbf{M}_{1(1)} + \mathbf{M}_{1(4)}][\mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)} \ \ x_w \mathbf{M}_{2(1)} + \mathbf{M}_{2(4)}]^{-1}$$

$$= [\mathbf{M}_{1(1)} \ \mathbf{M}_{1(2)} \ \mathbf{M}_{1(3)}][\mathbf{M}_{2(1)} \ \mathbf{M}_{2(2)} \ \mathbf{M}_{2(3)}]^{-1}$$

This is verified by brutal expansion of the matrixes elements in above equation and comparison among corresponding terms. We could resort to software with symbolic operation functionality to perform the verification; the piece of code for MATLAB is listed here:

```
%% CODE BEGIN
x=sym('x');
for r=1:3
```

```
for c=1:4
mOne(r,c) = sym(['mOne' num2str(r) num2str(c)]);
mTwo(r,c) = sym(['mTwo' num2str(r) num2str(c)]);
end
end
mOneX=[mOne(:,2), mOne(:,3), x*mOne(:,1)+mOne(:,4)];
mTwoX=[mTwo(:,2), mTwo(:,3), x*mTwo(:,1)+mTwo(:,4)];
T = simple( mOneX*inv(mTwoX) );
Tx = simple( limit(T,x,inf) );
R = simple( mOne(:,1:3)*inv(mTwo(:,1:3)) );
simple( Tx-R )
%% The output should be a matrix of zero.
%% CODE END
```

From *Formula II* we can know that (5-8) as $x_w \rightarrow \infty$. Since the form of (5-8) is symmetrical, we can verify that (5-8) also holds as $y_w \rightarrow \infty$ or $z_w \rightarrow \infty$. End. □

The meaning of *Formula III* can be interpreted in this way: given a 3D point which is far away enough from the two cameras, then the geometric relationship between its two image projections only depends on the perspective matrixes of the two cameras. In other words, given a pixel on one camera image, if the corresponding 3D point of this pixel is known to be far away from the two cameras, then even without any exact spatial information of this 3D point, the corresponding pixel on the other camera image can still be determined (suppose the perspective matrixes are known).

5.4 Approximate Estimation of the Visual Perception Depth using a 2D Laser Scanner

A prerequisite for performing the perspective transformation between the visual perceptions of the two vehicles is the knowledge of the visual perception depth. This knowledge can be estimated by stereo-vision, if correct correspondence is established (yet a challenging process) between the images pair in stereo-vision. Unfortunately, by so far, stereo-vision is not available in our vehicle sensor configurations. In spite of the absence of stereo-vision, approximate estimate of the visual perception depth can be obtained with the help of 2D range perception. Although the perspective transformation based on this approximate estimate might not be precise, its performance could still be lifelike enough for driver assistance, as will be shown later.

5.4.1 Coordinates Systems in an Intelligent Vehicle

Before further introduction of the approximate estimation method, we give some explanations on several coordinates systems that are concerned in an intelligent vehicle, namely the camera coordinates system (CCS), the laser scanner coordinates system (SCS), the ground coordinates system (GCS), and the vehicle coordinates system (VCS), as illustrated in Figure 5.4.

Camera Coordinates System: its origin and coordinate axes are denoted by $\{\mathbf{O_c},\mathbf{X_c},\mathbf{Y_c},\mathbf{Z_c}\}$; the origin $\mathbf{O_c}$ is situated at the virtual pinhole by which a projection line passes; the $\mathbf{O_c}$-$\mathbf{X_c}$-$\mathbf{Y_c}$ plane is parallel to the image plane.

Laser Scanner Coordinates System: its origin and coordinate axes are denoted by $\{\mathbf{O_s},\mathbf{X_s},\mathbf{Y_s},\mathbf{Z_s}\}$; the origin $\mathbf{O_s}$ is situated at the laser emitting point; the plane $\mathbf{Z_s}$=0 is the scanning plane of the 2D laser scanner.

Vehicle Coordinates System: (Let the vehicle be stationary on the ground plane) its origin and coordinate axes are denoted by $\{\mathbf{O_v},\mathbf{X_v},\mathbf{Y_v},\mathbf{Z_v}\}$; the coordinate axes $\{\mathbf{X_v},\mathbf{Y_v},\mathbf{Z_v}\}$ are respectively established along the longitudinal direction (pointing forward), along the lateral direction (pointing left), and along the vertical direction (point upward) of the vehicle; the origin $\mathbf{O_v}$ is at the projection of the rear wheel axle center on the ground.

Ground Coordinates System: (Let the vehicle be stationary on the ground plane) its origin and coordinate axes are denoted by $\{\mathbf{O_g},\mathbf{X_g},\mathbf{Y_g},\mathbf{Z_g}\}$; the origin $\mathbf{O_g}$ is at the projection of $\mathbf{O_c}$ on the ground, the axis $\mathbf{Z_g}$ points from $\mathbf{O_g}$ to $\mathbf{O_c}$ i.e. the ground plane is the plane $\mathbf{Z_g}$=0, and the axis $\mathbf{X_g}$ is along the projection of the axis $\mathbf{Z_c}$ on the ground.

It is worthy noting that these coordinates systems might be established differently; they are established in above way mainly for the convenience of calibration process and applications associated with intelligent vehicles.
The camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system, are always fixed with the vehicle (so they are also mutually fixed with each other). The rigid spatial relationships among them are calibrated off-line; a calibration method which reveals all these spatial relationships will be introduced in latter sub-section.

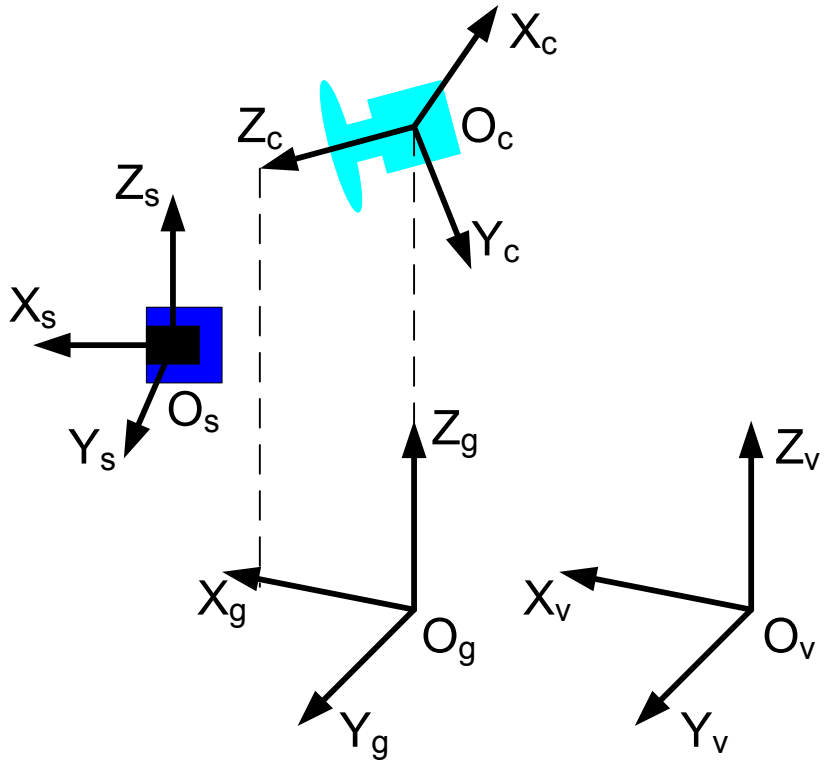


Figure 5.4 Coordinates systems in an intelligent vehicle

It is assumed that the $\mathbf{O_g}$-$\mathbf{X_g}$-$\mathbf{Y_g}$ plane and the $\mathbf{O_v}$-$\mathbf{X_v}$-$\mathbf{Y_v}$ plane are always situated on the ground surface during intelligent vehicle operation, i.e. the pitch-roll movements of the vehicle are assumed negligible. Although this assumption is not strictly correct, yet it is reasonable in practices.

5.4.2 Approximate Estimation of the Visual Perception Depth

The visual perception depth is approximately estimated in the following way.

First step, compute the depth i.e. the x_v value of the ground part on the visual perception using the *formula I* in Section 5.3.2. Suppose all the image part below the vanishing line corresponds to ground surface; then compute the depth of this part according to (5-6), considering that z_v=0. The vanishing line is determined by finding the solution (the set of $\mathbf{P_i}$) which makes the denominator in (5-6) be zero, i.e.

$$\mathbf{e}_3^{\mathrm{T}}[\mathbf{M}_{(1)} \quad \mathbf{M}_{(2)} \quad z_w\mathbf{M}_{(3)} + \mathbf{M}_{(4)}]^{-1}\mathbf{P_i} = 0$$

Second step, compute the depth of the image part that can be associated with 2D range perception. Given a range point, transformed into the vehicle coordinates system, denoted as $[x_{v(i)};y_{v(i)};z_{v(i)}]^T$. It is assumed that the detection of this point is caused by a vertical column which contains this point; the bottom part and the top part of this vertical column are respectively assumed to be situated on the ground surface and H_p meters (for example, let H_p be 2) above the ground surface; the width of this vertical column is chosen according to the range perception error. These assumptions are based on the following considerations: pedestrians, i.e. the environment objects that we care most concerning navigation safety, have vertical column-like shape normally within 2 meters height; besides, many other environment objects also have vertical shape, such as vehicles, traffic signs, building and trees.

The depth of all the image part that corresponds to this vertical column (determined through perspective mapping) is set to be $x_{v(i)}$. When the image projections of two such kind of vertical columns overlap with each other, the overlapping part will have two depth estimates, then only the smaller depth estimate is kept for the overlapping part. In other words, if an image part can be associated with two range perception points, then it is associated only with the point closer to the vehicle.

Third step, the image parts whose depth have not been determined yet are assumed to correspond to far-away objects and their depth is set to be *large*. This assumption seems rude but is reasonable: normally, these image parts either correspond to the objects that are beyond the detection range of the laser scanner or correspond to the objects high above the ground. If the former case holds true, this directly means that the objects are far away because the detection range of the laser scanner can be 80 meters. If the later case holds true, these objects high above the ground are anyway of less interest; the perspective transformation of these objects, even inaccurate, does not matter so much.

One example of the visual perception depth estimated through above steps is demonstrated in Figure 5.5, where the color varies from red to blue as the depth varies from small to large. In practice, it is not necessary to estimate the depth of the whole visual perception; the depth information of only the occluded part is needed. Besides, the Gaussian filtering is performed on the estimate of the visual perception depth in order to smooth visualization effect.

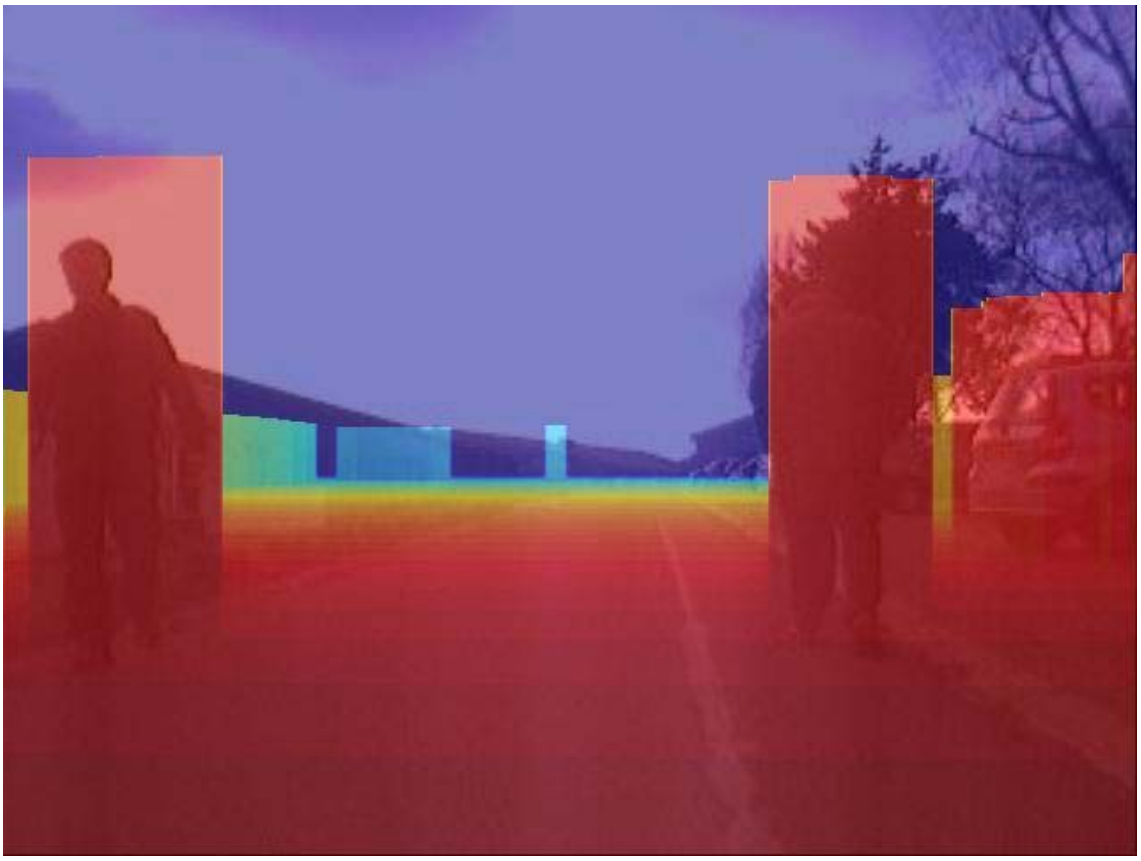


Figure 5.5 Approximate estimate of the visual perception depth

5.5 Perspective Transformation between the Visual Perceptions of Two Intelligent Vehicles

Let the relative pose between the front vehicle and the following vehicle be denoted as $[x_{12};y_{12};\theta_{12}]^T$. This relative pose is obtained using the indirect vehicle-to-vehicle relative pose estimation method introduced in Section 3.5). Suppose the perspective mappings between the vehicle coordinates systems and the image coordinates systems at the front vehicle and the following vehicle are respectively given by (calibrated off-line):

Vehicle 1: $\lambda_1 \mathbf{P_{i1(A)}} = \mathbf{M}_1 \cdot \mathbf{P_{v1(A)}}$

Vehicle 2: $\lambda_2 \mathbf{P_{i2(A)}} = \mathbf{M}_2 \cdot \mathbf{P_{v2(A)}}$

The $\mathbf{P_{i1(A)}}=[u_1,v_1,1]^T$ and $\mathbf{P_{i2(A)}}=[u_2,v_2,1]^T$ denote the augmented coordinates in the image coordinates systems respectively of the front and the following vehicles. The

$\mathbf{P}_{\mathbf{v}1(\mathrm{A})}=[x_{v1},y_{v1},z_{v1},1]^{\mathrm{T}}$ and $\mathbf{P}_{\mathbf{v}2(\mathrm{A})}=[x_{v2},y_{v2},z_{v2},1]^{\mathrm{T}}$ denote the augmented coordinates in the vehicle coordinates systems respectively of the front and the following vehicles. The relationship between $\mathbf{P}_{\mathbf{v}1(\mathrm{A})}$ and $\mathbf{P}_{\mathbf{v}2(\mathrm{A})}$ is given by:

$$\mathbf{P}_{\mathbf{v}2(\mathbf{A})} = \mathbf{RT}_{12} \cdot \mathbf{P}_{\mathbf{v}1(\mathbf{A})}$$

$$\mathbf{RT}_{12} = \begin{bmatrix} \cos\theta_{12} & -\sin\theta_{12} & 0 & x_{12} \\ \sin\theta_{12} & \cos\theta_{12} & 0 & y_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The visual perceptions of the two vehicles can be related to a common spatial reference which is chosen to be the second vehicle coordinates system:

Vehicle 1: $\lambda_1 \mathbf{P}_{\mathbf{i}1(\mathbf{A})} = (\mathbf{M}_1 \cdot \mathbf{RT}_{12}^{-1})\mathbf{P}_{\mathbf{v}2(\mathbf{A})}$

Vehicle 2: $\lambda_2 \mathbf{P}_{\mathbf{i}2(\mathbf{A})} = \mathbf{M}_2 \cdot \mathbf{P}_{\mathbf{v}2(\mathbf{A})}$

Following the previous sub-section, the visual perception consists of two parts: the depth estimate of one part is determined; the depth estimate of the other part is *large* which means that this part corresponds to far-away objects. For the former part, perspective transformation between the visual perceptions of the two vehicles is carried out based on *formula II* in section 5.3.2. For the latter part, perspective transformation is carried out based on *formula III*.

5.6 Extrinsic Co-Calibration of a Camera and a 2D Laser Scanner

For the previously introduced method to be realized, the extrinsic parameters which characterize the (rigid) spatial relationships among the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system, have to be calibrated off-line. In this sub-section, we introduce a COMPREHENSIVE extrinsic calibration method which reveals ALL these spatial relationships, based only on the popular chessboard calibration practice [Zhang & Pless 2004] with few extra measurements.

5.6.1 Mathematical Fundaments and Denotations

In Section 5.4.1, we have introduced the camera coordinates system (CCS), the laser scanner coordinates system (SCS), the ground coordinates system (GCS), and the vehicle coordinates system (VCS) concerning an intelligent vehicle. As we use a chessboard for the calibration, we would like to introduce one more sort of coordinates system, i.e. the chessboard coordinates system (PCS), as illustrated in Figure 5.6.

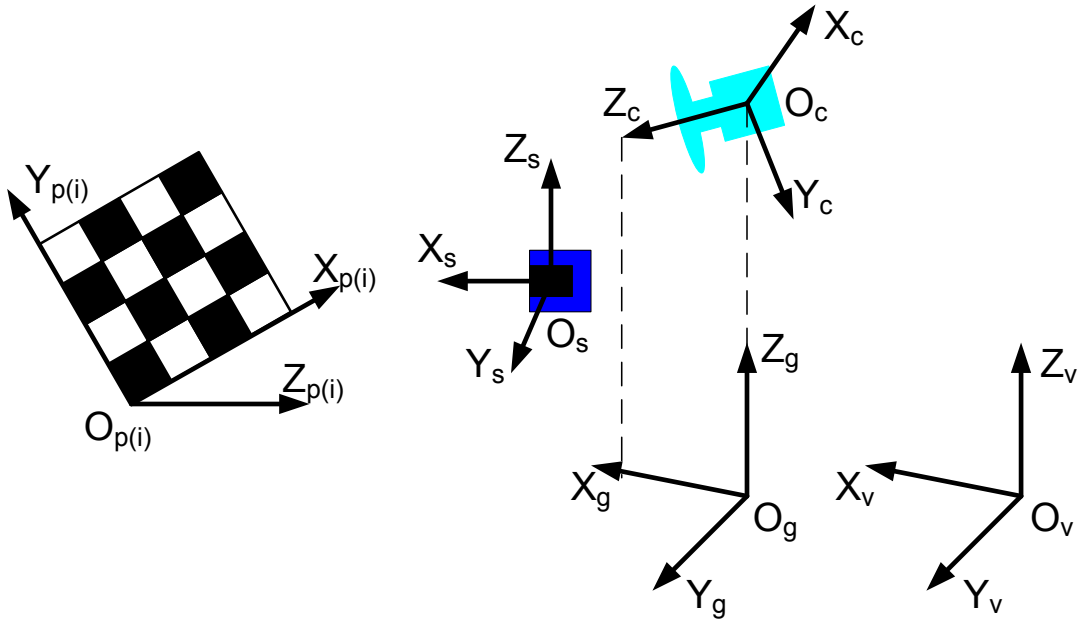


Figure 5.6 Coordinates systems concerned in the calibration

Given a pose of the chessboard plane, the origin and the coordinate axes of the PCS are denoted by $\{\mathbf{O_p},\mathbf{X_p},\mathbf{Y_p},\mathbf{Z_p}\}$, where the plane $\mathbf{Z_p}=0$ is situated on the chessboard plane, the origin $\mathbf{O_p}$ is situated at the left-bottom corner of the chessboard, the axis $\mathbf{X_p}$ is along the bottom edge of the chessboard, and the axis $\mathbf{Y_p}$ is along the left edge of the chessboard, as illustrated in Figure 5.6. The chessboard is placed with several different poses in the perception field of the camera and the 2D laser scanner; for each pose, a sub-script '$_{(i)}$' is used to distinguish the PCS. Thus the different chessboard poses that are used for calibration are denoted by a set of $PCS_{(i)}$, i.e. $\{\mathbf{O_{p(1)}},\mathbf{X_{p(1)}},\mathbf{Y_{p(1)}},\mathbf{Z_{p(1)}}\}$, $\{\mathbf{O_{p(2)}},\mathbf{X_{p(2)}},\mathbf{Y_{p(2)}},\mathbf{Z_{p(2)}}\}$ *etc*.

The $\{\mathbf{X_a},\mathbf{Y_a},\mathbf{Z_a}\}$ also denote the unit vectors along corresponding coordinate axes ($\mathbf{a}$={c, s, g, v, p(1), p(2), …}). The capitalized letter **R** and **T** generally denote a 3x3 rotation matrix and a 3x1 translation vector respectively. The $\mathbf{R_{ab}}$ and $\mathbf{T_{ab}}$ respectively denote

the rotation and translation from the coordinates system $\{\mathbf{O_a},\mathbf{X_a},\mathbf{Y_a},\mathbf{Z_a}\}$ ($\mathbf{a}$={c, s, g, v, p(1), p(2), …}) to the coordinate system $\{\mathbf{O_b},\mathbf{X_b},\mathbf{Y_b},\mathbf{Z_b}\}$ ($\mathbf{b}$={c, s, g, v, p(1), p(2), …}). For example, $\mathbf{R_{cs}}$ and $\mathbf{T_{cs}}$ denote the transformation from the CCS to the SCS. The capitalized letter **M** generally denotes a point and $\mathbf{M_a}=[x_a,y_a,z_a]^T$ denotes the coordinates of **M** in the coordinate system $\{\mathbf{O_a},\mathbf{X_a},\mathbf{Y_a},\mathbf{Z_a}\}$ ($\mathbf{a}$={c, s, g, v, p(1), p(2), …}). Thus the following relationships hold:

$$\begin{array}{lll} & \mathbf{M_b} = \mathbf{R_{ab}}\mathbf{M_a} + \mathbf{T_{ab}} & \\ \text{Dual relationship} & \mathbf{R_{ba}} = \mathbf{R_{ab}}^T & \mathbf{T_{ba}} = -\mathbf{R_{ab}}^T\mathbf{T_{ab}} \\ \text{Chain relationship} & \mathbf{R_{ab}} = \mathbf{R_{fb}}\mathbf{R_{af}} & \mathbf{T_{ab}} = \mathbf{R_{fb}}\mathbf{T_{af}} + \mathbf{T_{fb}} \end{array} \quad (5\text{-}9)$$

The $\mathbf{a},\mathbf{b},\mathbf{f}$={c, s, g, v, p(1), p(2), …}. The spatial relationships among the CCS, the SCS, the GCS, and the VCS are fixed; the objective of the extrinsic calibration is to reveal these spatial relationships, i.e. the extrinsic parameters $\mathbf{R_{ab}}$ and $\mathbf{T_{ab}}$ ($\mathbf{a},\mathbf{b}$={c, s, g, v}).
In the CCS, the '$\mathbf{N_{p(i)c}}$' is used to denote the perpendicular vector from the $\mathbf{O_c}$ to the chessboard plane $PCS_{(i)}$ $\{\mathbf{O_{p(i)}}, \mathbf{X_{p(i)}}, \mathbf{Y_{p(i)}}, \mathbf{Z_{p(i)}}\}$, where the magnitude $\|\mathbf{N_{p(i)c}}\|$ equals the distance from the $\mathbf{O_c}$ to the $PCS_{(i)}$. The '$\mathbf{N_{gc}}$' is used to denote the perpendicular vector from the $\mathbf{O_c}$ to the ground plane. Let $\mathbf{N_{Gc}}=[\mathbf{N_{gc}}^T,n_0]^T$ and let the ground plane be represented by equation $\mathbf{N_{Gc}}^T[\mathbf{M_c}^T,1]^T=0$.
Let $\mathbf{e_1}$, $\mathbf{e_2}$, $\mathbf{e_3}$ respectively denote the constant vectors $[1,0,0]^T$, $[0,1,0]^T$ and $[0,0,1]^T$. The norm '||.||' denotes the L2-norm, i.e. given an arbitrary vector **V**, $\|\mathbf{V}\|^2=\mathbf{V^TV}$.

5.6.2 Comprehensive Extrinsic Calibration Method: Basic Version

The proposed calibration method consists of three parts: 1) the calibration between the CCS and SCS, using the method introduced in [Zhang & Pless 2004] that is based on the chessboard calibration practice; 2) the calibration between the CCS and the GCS, based on the same chessboard calibration practice; 3) the calibration between the GCS and the VCS, with the help of few extra measurements in addition to the chessboard calibration practice.

Part I: the calibration between the CCS and the SCS

Assume that the intrinsic parameters of the camera are already calibrated using the method introduced in [Zhang 2000]; given several chessboard poses that are used for calibration: $PCS_{(1)}\{\mathbf{O_{p(1)}},\mathbf{X_{p(1)}},\mathbf{Y_{p(1)}},\mathbf{Z_{p(1)}}\}$, $PCS_{(2)}\{\mathbf{O_{p(2)}},\mathbf{X_{p(2)}},\mathbf{Y_{p(2)}},\mathbf{Z_{p(2)}}\}$ *etc*. For a pose $PCS_{(i)}\{\mathbf{O_{p(i)}},\mathbf{X_{p(i)}},\mathbf{Y_{p(i)}},\mathbf{Z_{p(i)}}\}$, a 3-vector $\mathbf{N_{p(i)c}}$ that is perpendicular to the chessboard plane is computed as:

$$\mathbf{N_{p(i)c}} = (\mathbf{e_3}^{\mathbf{T}}\mathbf{R_{p(i)c}}^{\mathbf{T}}\mathbf{T_{p(i)c}})\mathbf{R_{p(i)c}}\mathbf{e_3}$$

where the rotation matrix $\mathbf{R_{p(i)c}}$ and the translation vector $\mathbf{T_{p(i)c}}$ are computed based on the homography between the plane $Z_{p(i)}$=0 of the $PCS_{(i)}$ and the 2D image coordinates system [Zhang 2000].

According to the geometric constraint that laser points should be located on the chessboard plane, the relative pose i.e. $\mathbf{R_{cs}}$ and $\mathbf{T_{cs}}$ between the camera and the 2D laser scanner are optimized by minimizing the summed square of distances (SSD) of all the laser points to corresponding chessboard planes:

$$\begin{aligned} &\{\mathbf{R_{cs}}, \mathbf{T_{cs}}\} = \arg\min_{\mathbf{R_{cs}}, \mathbf{T_{cs}}} F_1 \\ &F_1 = \sum_i \sum_j [\frac{\mathbf{N_{p(i)c}}}{\| \mathbf{N_{p(i)c}} \|} \bullet \mathbf{R_{cs}}^{-1}(\mathbf{M_{s(i,j)}} - \mathbf{T_{cs}}) - \| \mathbf{N_{p(i)c}} \|]^2 \end{aligned} \tag{5-10}$$

where the rotation matrix $\mathbf{R_{cs}}$ is parameterized by a 3-vector using the Rodrigues formula [Faugeras 1993]; $\mathbf{M_{s(i,j)}}$ is the j-th laser point on the $PCS_{(i)}$. The initial value of $\mathbf{R_{cs}}$ and $\mathbf{T_{cs}}$ are estimated by solving a linear equation problem [Zhang & Pless 2004]:

Initial Estimation of $\mathbf{R_{cs}}$ and $\mathbf{T_{cs}}$:

According to the geometric constraint that laser points should be located on the chessboard plane, the following relationship holds:

$$\mathbf{N_{p(i)c}} \bullet \mathbf{R_{cs}}^{-1}(\mathbf{M_{s(i,j)}} - \mathbf{T_{cs}}) = \| \mathbf{N_{p(i)c}} \|^2$$

Since all the laser points are on the plane Z_s=0 in the SCS, above equation can be rewritten as follows:

$$\mathbf{N_{p(i)c}} \cdot \mathbf{H} \begin{bmatrix} \mathbf{M_{s(i,j)}} \\ 1 \end{bmatrix} = \| \mathbf{N_{p(i)c}} \|^2$$

$$\mathbf{H} = \mathbf{R_{cs}}^{-1}[\mathbf{e_1} \quad \mathbf{e_2} \quad -\mathbf{T_{cs}}]$$

Where the $\mathbf{H}$ is a 3x3 transform matrix from the SCS to the CCS; for each laser point $\mathbf{M_{s(i,j)}}$ and corresponding chessboard pose $PCS_{(i)}$, a linear equation of the unknown parameters of $\mathbf{H}$ can be formed. For all the laser points and corresponding chessboard poses, a linear equation group of the unknown parameters of $\mathbf{H}$ can be formed and can be solved with linear least squares method. Once the $\mathbf{H}$ is computed, the $\mathbf{R_{cs}}$ and $\mathbf{T_{cs}}$ can be estimated as follows:

$$\mathbf{R_{cs}} = \left[\mathbf{H}\cdot\mathbf{e_1} \quad \mathbf{H}\cdot\mathbf{e_2} \quad (\mathbf{H}\cdot\mathbf{e}_1)\times(\mathbf{H}\cdot\mathbf{e}_2)\right]^{\mathbf{T}}$$

$$\mathbf{T_{cs}} = -\left[\mathbf{H}\cdot\mathbf{e_1} \quad \mathbf{H}\cdot\mathbf{e_2} \quad (\mathbf{H}\cdot\mathbf{e}_1)\times(\mathbf{H}\cdot\mathbf{e}_2)\right]^{\mathbf{T}}(\mathbf{H}\cdot\mathbf{e_3})$$

The computed matrix $\mathbf{R_{cs}}$ might not satisfy the orthonormal condition of a rotation matrix; it can be approximated by a rotation matrix $\mathbf{R_{cs}}^*$ which minimizes the Frobenius norm of $\mathbf{R_{cs}}^*-\mathbf{R_{cs}}$ (details can be referred to [Golub & Loan 1996]). End. □

The $\mathbf{R_{sc}}$ and $\mathbf{T_{sc}}$ can be computed using the dual relationship: $\mathbf{R_{sc}}=\mathbf{R_{cs}}^{\mathbf{T}}$; $\mathbf{T_{sc}}=-\mathbf{R_{cs}}^{\mathbf{T}}\mathbf{T_{cs}}$.

Part II: the calibration between the CCS and the GCS

We introduce how to reveal the spatial relationship between the CCS and the GCS, based on the same chessboard calibration practice (as used for the calibration between the CCS and the SCS) without any extra calibration practice.

In the original chessboard calibration practice as in [Zhang & Pless 2004], one can hold the chessboard either on the ground or in the air, only if the camera and the 2D laser scanner can both perceive the chessboard; yet it is more convenient to hold the chessboard on the ground than in the air. Holding the chessboard on the ground will not have essential influence on the calibration results; the reason can be understood in this way: suppose one holds the chessboard in the air, imagine that the chessboard is extended onto the ground, thus holding the chessboard in the air is just like holding an extended chessboard on the ground.

A calibration field where the ground is fairly flat could always be found; for example, on the floor in a garage room, or on a locally flat road. Then, posing the chessboard on the ground brings one more geometric constraint, which is referred to as *ground plane constraint* here; it means that the bottom edge of the chessboard is situated on the ground plane, or in geometric terms, the line $\mathbf{O_{p(i)}}+\lambda\mathbf{X_{p(i)}}$ (λ is a scalar) is situated on the ground plane. The ground plane constraint not only helps reveal the spatial relationship between the CCS and GCS, but also helps refine the calibration results.

Let l be the length of the chessboard bottom edge; the points $\mathbf{O_{p(i)}}$ and $\mathbf{O_{p(i)}}+l\cdot\mathbf{X_{p(i)}}$, which represent the two corner points on the chessboard bottom edge, are chosen as control points. The relative pose between the $PCS_{(i)}$ and the camera, i.e. $\mathbf{R_{p(i)c}}$ and $\mathbf{T_{p(i)c}}$ can be computed, as mentioned in section 3.1; in the CCS, the coordinates of $\mathbf{O_{p(i)}}$ and $\mathbf{X_{p(i)}}$ are respectively $\mathbf{T_{p(i)c}}$ and $\mathbf{R_{p(i)c}e_1}$. As the ground plane is denoted by equation $\mathbf{N_{Gc}}^{\mathrm{T}}[\mathbf{M_c}^{\mathrm{T}},1]^{\mathrm{T}}=0$, a linear equation can be established:

$$\mathbf{G}\mathbf{N}_{\mathbf{Gc}} = \mathbf{0}$$

$$\mathbf{G} = \begin{bmatrix} \cdots & \cdots \\ \mathbf{T}_{\mathbf{p(i)c}}{}^{\mathbf{T}} & 1 \\ \mathbf{T}_{\mathbf{p(i)c}}{}^{\mathbf{T}} + l \cdot \mathbf{e_1}{}^{\mathbf{T}}\mathbf{R}_{\mathbf{p(i)c}}{}^{\mathbf{T}} & 1 \\ \cdots & \cdots \end{bmatrix} \tag{5-11}$$

The solution of this homogeneous linear equation, i.e. $\mathbf{N_{Gc}}$ is the eigenvector associated with the smallest eigenvalue of $\mathbf{G^TG}$. Let $\mathbf{N_{Gc}}$ be decomposed as $\mathbf{N_{Gc}}=[\mathbf{N_{gc}}^T, n_0]^T$; the 3-vector $\mathbf{N_{gc}}$ is perpendicular to the ground plane. According to the establishment of the GCS as specified in Section 5.4.1, the spatial relationship between the CCS and GCS i.e. $\mathbf{R_{gc}}$ and $\mathbf{T_{gc}}$, is computed as follows:

$$\begin{aligned} \mathbf{T}_{\mathbf{gc}} &= -\frac{n_0}{\| \mathbf{N}_{\mathbf{gc}} \|^2}\mathbf{N}_{\mathbf{gc}} \\ \mathbf{R}_{\mathbf{gc}}\mathbf{e_1} &= (-\frac{\mathbf{N}_{\mathbf{gc}}{}^{\mathbf{T}}\mathbf{e_3}}{\| \mathbf{N}_{\mathbf{gc}} \|^2}\mathbf{N}_{\mathbf{gc}} + \mathbf{e_3}) / \| -\frac{\mathbf{N}_{\mathbf{gc}}{}^{\mathbf{T}}\mathbf{e_3}}{\| \mathbf{N}_{\mathbf{gc}} \|^2}\mathbf{N}_{\mathbf{gc}} + \mathbf{e_3} \| \\ \mathbf{R}_{\mathbf{gc}}\mathbf{e_3} &= -\mathbf{T}_{\mathbf{gc}} / \| \mathbf{T}_{\mathbf{gc}} \| \\ \mathbf{R}_{\mathbf{gc}}\mathbf{e_2} &= (\mathbf{R}_{\mathbf{gc}}\mathbf{e_3}) \times (\mathbf{R}_{\mathbf{gc}}\mathbf{e_1}) \end{aligned} \tag{5-12}$$

Lemma: Given a plane denoted as $\mathbf{N_p}^T[\mathbf{M}^T,1]^T=0$, where $\mathbf{N_p}=[\mathbf{N}^T,n_0]^T$ and $\mathbf{N}$ is a 3-vector; for an arbitrary point $\mathbf{M_a}$, the projection of $\mathbf{M_a}$ on this plane, denoted as $\mathbf{M_{a(p)}}$, is computed as:

$$\mathbf{M}_{\mathbf{a(p)}} = -\frac{\mathbf{N}^{\mathbf{T}}\mathbf{M}_{\mathbf{a}} + n_0}{\| \mathbf{N} \|^2}\mathbf{N} + \mathbf{M}_{\mathbf{a}}$$

Proof: As the 3-vector $\mathbf{N}$ is perpendicular to the plane, the projection $\mathbf{M_{a(p)}}$ is in the form $\mathbf{M_{a(p)}}=\mathbf{M_a}+\lambda\mathbf{N}$ where λ is a scalar to-be-computed. Substitute $\mathbf{M_{a(p)}}=\mathbf{M_a}+\lambda\mathbf{N}$ for $\mathbf{M}$ in the equation $\mathbf{N_p}^T[\mathbf{M}^T,1]^T=0$, i.e. $\mathbf{N}^T(\mathbf{M_a}+\lambda\mathbf{N})+n_0=0$, and compute the λ:

$$\lambda = -\frac{\mathbf{N}^{\mathbf{T}}\mathbf{M}_{\mathbf{a}} + n_0}{\mathbf{N}^{\mathbf{T}}\mathbf{N}} = -\frac{\mathbf{N}^{\mathbf{T}}\mathbf{M}_{\mathbf{a}} + n_0}{\| \mathbf{N} \|^2}$$

Substitute the λ into $\mathbf{M_{a(p)}}=\mathbf{M_a}+\lambda\mathbf{N}$ and the lemma is done. □

Proof of (5-12): In the CCS, let the ground plane be denoted by equation $\mathbf{N_{Gc}}^{\mathbf{T}}[\mathbf{M_c}^{\mathrm{T}},1]^{\mathbf{T}}=0$ and the $\mathbf{N_{Gc}}=[\mathbf{N_{gc}}^{\mathrm{T}},n_0]^{\mathrm{T}}$. According to the establishment of the GCS as specified in section 2, the $\mathbf{O_g}$ (i.e. $\mathbf{T_{gc}}$ in the CCS) is the projection of the $\mathbf{O_c}$ (i.e. $\mathbf{0}$ in the CCS) on the ground plane; then $\mathbf{T_{gc}}$ can be computed via the lemma:

$$\mathbf{T_{gc}} = -\frac{\mathbf{N_{gc}}^{\mathbf{T}}\mathbf{0}+n_0}{\|\mathbf{N_{gc}}\|^2}\mathbf{N_{gc}} + \mathbf{0} = -\frac{n_0}{\|\mathbf{N_{gc}}\|^2}\mathbf{N_{gc}}$$

As the axis $\mathbf{Z_g}$ points from $\mathbf{O_g}$ to $\mathbf{O_c}$, the unit vector $\mathbf{Z_g}$ (i.e. $\mathbf{R_{gc}e_3}$ in the CCS) is computed as:

$$\mathbf{R_{gc}e_3} = -\mathbf{T_{gc}} / \|\mathbf{T_{gc}}\|$$

Select a point on the axis $\mathbf{Z_c}$ (let it be $\mathbf{e_3}$ in the CCS) and compute its projection on the ground plane:

$$\mathbf{P_z} = -\frac{\mathbf{N_{gc}}^{\mathbf{T}}\mathbf{e_3}+n_0}{\|\mathbf{N_{gc}}\|^2}\mathbf{N_{gc}} + \mathbf{e_3}$$

As the axis $\mathbf{X_g}$ is along the projection of the axis $\mathbf{Z_c}$ on the ground, the unit vector $\mathbf{X_g}$ (i.e. $\mathbf{R_{gc}e_1}$ in the CCS) is computed as:

$$\mathbf{R_{gc}e_1} = (\mathbf{P_z} - \mathbf{T_{gc}}) / \|\mathbf{P_z} - \mathbf{T_{gc}}\|$$

$$= (-\frac{\mathbf{N_{gc}}^{\mathbf{T}}\mathbf{e_3}}{\|\mathbf{N_{gc}}\|^2}\mathbf{N_{gc}} + \mathbf{e_3}) / \|-\frac{\mathbf{N_{gc}}^{\mathbf{T}}\mathbf{e_3}}{\|\mathbf{N_{gc}}\|^2}\mathbf{N_{gc}} + \mathbf{e_3}\|$$

According to the right-hand rule, the unit vector $\mathbf{Y_g}$ (i.e. $\mathbf{R_{gc}e_2}$ in the CCS) is computed as:

$$\mathbf{R_{gc}e_2} = (\mathbf{R_{gc}e_3}) \times (\mathbf{R_{gc}e_1})$$

End. □

The $\mathbf{R_{cg}}$ and $\mathbf{T_{cg}}$ can be computed using the dual relationship: $\mathbf{R_{cg}}=\mathbf{R_{gc}}^{\mathbf{T}}$; $\mathbf{T_{cg}}=-\mathbf{R_{gc}}^{\mathbf{T}}\mathbf{T_{gc}}$. The spatial relationship between the SCS and the GCS can be computed using the chain relationship: $\mathbf{R_{sg}}=\mathbf{R_{cg}}\mathbf{R_{sc}}$; $\mathbf{T_{sg}}=\mathbf{R_{cg}}\mathbf{T_{sc}}+\mathbf{T_{cg}}$.

Part III: the calibration between the GCS and the VCS

The chessboard calibration practice is enough for revealing spatial relationships among the CCS, the SCS, and the GCS as introduced previously. In order to further relate the CCS, the SCS, and the GCS to the VCS, an extra step of registering few control points in the VCS is needed and it is carried out by manual measurements.

According to the establishment of the GCS and the VCS as mentioned in Section 5.4.1, the transformation between the GCS and VCS is given by a rotation around the axis Z_g and a translation along the ground plane, as follows:

$$\begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ 0 \end{bmatrix}$$

Given a chessboard pose $PCS_{(i)}$, the origin $\mathbf{O_{p(i)}}$ is chosen as a control point, which is called the *ground control point*. The coordinates of $\mathbf{O_{p(i)}}$ in the GCS is computed as: $\mathbf{O_{p(i)g}}=\mathbf{R_{cg}T_{p(i)c}}+\mathbf{T_{cg}}$. Choose a set of ground control points $\mathbf{O_{p(i)}}$ (it is NOT necessary to choose the origins of all chessboard poses), compute their coordinates $\mathbf{O_{p(i)g}}=[x_{og(i)},y_{og(i)}]^T$ in the GCS, and manually measure their coordinates $\mathbf{O_{p(i)v}}=[x_{ov(i)},y_{ov(i)}]^T$ in the VCS. Since $z_v=z_g$ always holds here, the third coordinate is omitted.

The objective is to reveal the parameters $\{\theta,t_x,t_y\}$ from the set of coordinates pairs $\{[x_{og(i)},y_{og(i)}]^T, [x_{ov(i)},y_{ov(i)}]^T\}$; each coordinates pair corresponds to a ground control point. Two coordinates pairs are enough to determine the parameters $\{\theta,t_x,t_y\}$, while more available coordinates pairs might be expected to yield more accurate results. The initial value of $\{\theta,t_x,t_y\}$ is estimated by solving the following linear equation:

$$\begin{bmatrix} \dots & \dots & \dots & \dots \\ x_{og(i)} & -y_{og(i)} & 1 & 0 \\ y_{og(i)} & x_{og(i)} & 0 & 1 \\ \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} \cos\theta \\ \sin\theta \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} \dots \\ x_{ov(i)} \\ y_{ov(i)} \\ \dots \end{bmatrix}$$

Afterward, an iteration process is carried out. At each iteration step, the non-linear function '$\cos\theta$' and '$\sin\theta$' are locally linearized with last estimate of θ; the increment of θ and new $\{t_x,t_y\}$ are computed by solving a linear equation:

$$\begin{bmatrix} \dots & \dots & \dots \\ -x_{og(i)}\sin\theta_{k-1} - y_{og(i)}\cos\theta_{k-1} & 1 & 0 \\ x_{og(i)}\cos\theta_{k-1} - y_{og(i)}\sin\theta_{k-1} & 0 & 1 \\ \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} \Delta\theta_k \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} \dots \\ x_{ov(i)} - x_{og(i)}\cos\theta_{k-1} + y_{og(i)}\sin\theta_{k-1} \\ y_{ov(i)} - x_{og(i)}\sin\theta_{k-1} - y_{og(i)}\cos\theta_{k-1} \\ \dots \end{bmatrix}$$

Normally, the results of $\{\theta,t_x,t_y\}$ will converge after several iterations; then the rotation $\mathbf{R_{gv}}$ and the translation $\mathbf{T_{gv}}$ are obtained. Since the spatial relationships among the CCS, the SCS, and the GCS have already been calibrated using the method described previously, by so far, all the spatial relationships among the CCS, the SCS, the GCS, and the VCS can be derived using the dual relationship and the chain relationship in (5-9). Therefore, the comprehensive extrinsic calibration of the camera and the 2D laser scanner is performed.

5.6.3 Comprehensive Extrinsic Calibration Method: Improved Versions

The basic version of the comprehensive extrinsic calibration method is introduced in the previous sub-section. Its performance depends on the accuracy of the camera intrinsic parameters which are not precisely known in practice. Concerning the calibration between the camera and the 2D laser scanner as in [Zhang & Pless 2004], besides the basic method as reviewed in *Part I* in Section 5.6.2, [Zhang & Pless 2004] further proposes a *global optimization* strategy which optimizes not only the $\{\mathbf{R_{cs}}, \mathbf{T_{cs}}\}$ but also the $\{\mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}\}$ ($\mathbf{A}$ is the camera intrinsic matrix) in a joint objective function:

$$\begin{aligned}
&\{\mathbf{R_{cs}}, \mathbf{T_{cs}}, \mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}\} = \underset{\mathbf{R_{cs}},\mathbf{T_{cs}},\mathbf{A},\mathbf{R_{p(i)c}},\mathbf{T_{p(i)c}}}{\arg\min} F_2 \\
&F_2 = \sum_i \sum_j [\frac{\mathbf{N_{p(i)c}}}{\| \mathbf{N_{p(i)c}} \|} \cdot \mathbf{R_{cs}}^{-1}(\mathbf{M_{s(i,j)}} - \mathbf{T_{cs}}) - \| \mathbf{N_{p(i)c}} \|]^2 \\
&\quad + \alpha \sum_i \sum_k \| \mathbf{m_{(i,k)}} - \mathbf{m}(\mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}, \mathbf{M_{p(i,k)}}) \|^2
\end{aligned} \tag{5-13}$$

where $\mathbf{m_{(i,k)}}$ is the extracted image coordinates of the k-th control point for the $PCS_{(i)}$; $\mathbf{m}(\mathbf{A},\mathbf{R_{p(i)c}},\mathbf{T_{p(i)c}},\mathbf{M_{p(i,k)}})$ is its projected image coordinates. This global optimization strategy can be used to refine the calibration results as presented in [Zhang & Pless 2004]; it can be incorporated into the basic version of the comprehensive extrinsic calibration method to refine the estimates of all the spatial relationships among the CCS, the SCS, the GCS, and the VCS. Therefore, an improved version of the comprehensive extrinsic calibration method is formed, which is called the *improved version I* in this dissertation.

On one hand, the global optimization strategy in [Zhang & Pless 2004] refines the estimates of $\{\mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}\}$; on the other hand, it over-adjusts the estimates slightly to fit them to the sensor measurements that are also affected by noises. To make the global optimization strategy more reasonable, the *ground plane constraint* introduced in *Part*

II in Section 5.6.2 is proposed to be taken into account as a term of the objective function, i.e. the last term of F_3, which stands for the summed square of distances of all the $\mathbf{O}_{\mathbf{p(i)}}$ and $\mathbf{O}_{\mathbf{p(i)}}+l\cdot\mathbf{X}_{\mathbf{p(i)}}$ to the ground plane, as follows:

$$
\begin{aligned}
&\{\mathbf{R_{cs}}, \mathbf{T_{cs}}, \mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}, \mathbf{N_{Gc}}\} = \underset{\mathbf{R_{cs}}, \mathbf{T_{cs}}, \mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}}{\arg\min} F_3 \\
&F_3 = \sum_i \sum_j [\frac{\mathbf{N_{p(i)c}}}{\| \mathbf{N_{p(i)c}} \|} \cdot \mathbf{R_{cs}}^{-1} (\mathbf{M_{s(i,j)}} - \mathbf{T_{cs}}) - \| \mathbf{N_{p(i)c}} \|]^2 \\
&\quad + \alpha \sum_i \sum_k \| \mathbf{m_{(i,k)}} - \mathbf{m}(\mathbf{A}, \mathbf{R_{p(i)c}}, \mathbf{T_{p(i)c}}, \mathbf{M_{p(i,k)}}) \|^2 \\
&\quad + \beta \sum_i \frac{\| [\mathbf{T_{p(i)c}}; 1] \mathbf{N_{Gc}} \|^2 + \| [\mathbf{T_{p(i)c}} + l \cdot \mathbf{R_{p(i)c}} \mathbf{e_1}; 1] \mathbf{N_{Gc}} \|^2}{\| \mathbf{N_{Gc}} \|^2}
\end{aligned}
\tag{5-14}
$$

During the optimization, the initial value of the $\mathbf{N_{Gc}}$ is computed according to the initial estimates of {$\mathbf{A}$, $\mathbf{R_{p(i)c}}$, $\mathbf{T_{p(i)c}}$}, using the method described in section 3.2. The Levenberg-Marquardt method [More 1977] is used as the optimization technique for all above optimization processes. The α is a scalar weight which normalizes the relative contribution of the laser error term and the camera error term [Zhang & Pless 2004]. The β is a scalar weight which in practice can be set to a comparatively large value; in our implementation, it is set to be 100. This proposed optimization strategy with the ground plane constraint is incorporated into the basic version of the comprehensive extrinsic calibration method, thus forming another improved version of the method which is called the *improved version II* in this dissertation.

5.7 Summary

We have specified the front-following vehicles scenario to which the proposed idea of cooperative augmented reality is applied. We have reviewed the pinhole camera model and described how to establish spatial relationship between two views (easily extendable to multi-views case) according to perspective geometry. We have described several coordinates systems i.e. the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system that are concerned in an intelligent vehicle; we have introduced a technique of utilizing a 2D laser scanner to assist a mono-camera in estimating the visual perception depth approximately. We have presented how to map the visual perception of a vehicle onto that of another vehicle, abiding by the multi-views perspective geometry described. We

have also introduced a new extrinsic calibration method for a camera and a 2D laser scanner, which can reveal all the spatial relationships among the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system, based only on the popular chessboard calibration practice with few extra measurements.

CHAPTER 6 Implementation and Experimentation

Résumé

Nous avons présenté les conditions expérimentales et les résultats expérimentaux concernant la localisation coopérative, la cartographie locale coopérative et la réalité augmentée coopérative. Nous avons présenté les résultats d'une étude comparative fondée sur la simulation qui démontre l'avantage de l'architecture de localisation coopérative proposée utilisant le filtre SCIF (l'approche *SCIFCL*), notamment pour les véhicules intelligents avec des capacités de positionnement absolu hétérogènes. Un avantage important de la méthode SCIFCL est qu'elle assure une localisation améliorée naturellement répartie au sein du réseau de véhicules, tout en gardant une consistance raisonnable pour l'estimation de l'état de chaque véhicule. Nous avons également présenté les résultats de tests réels sur la localisation coopérative, qui conduisent à des conclusions similaires à l'étude comparative fondée sur la simulation. Nous avons démontré les performances de la méthode de fusion de grilles occupations, fondés sur des tests effectués avec des données réelles. En dépit d'une erreur initiale intentionnellement exagérée, les cartes locales construites par différents véhicules peuvent toujours être agrégées correctement en utilisant la méthode proposée. D'ailleurs, la méthode de fusion des grilles d'occupation a le potentiel de trouver une solution pour le problème dit de « kidnapping ». Nous avons démontré les performances de la méthode baptisée comme « *réalité augmentée coopérative* », qui réalise un effet vif de 'voir' à travers le véhicule leader pour le véhicule suiveur dans le scénario de véhicules « leader-suiveur ».

6.1 Introduction

In this chapter, we describe concrete implementation and integration of the proposed methods on our experimental vehicle platforms and demonstrate the experimental results concerning cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality.

6.2 Cooperative Localization using Split Covariance Intersection Filter

6.2.1 Simulation Based Comparative Study

We carried out simulation based experiments on the proposed architecture of cooperative localization using split covariance intersection filter. Simulation was chosen for the tests mainly for two reasons: First, in simulation, we can examine the pure performance of a cooperative localization architecture, which is exempted from the influences of *ad hoc* implementation factors. Second, in simulation, we can easily tune the tests conditions, some of which are not easy to be satisfied in real-data tests. For example, we can set the number of vehicles to be as many as we like. It is true that the gap between the simulation performance and the performance in reality does always exist, yet simulation can demonstrate the reasonableness and potential of a method and serve as a guide for real implementation.

As the experiments are carried out in simulation, a comparative study could be more meaningful than only demonstrating the performance of the proposed method. Therefore, the proposed cooperative localization method and several other methods are executed simultaneously on the same synthetic data and their respective performances are compared. The methods under tests are as follows:

Single Vehicle Localization Method [Rezaei & Sengupta 2007] (SL):

Each ego-vehicle performs localization using only its own sensor data and using the EKF for data fusion. More specifically, at each period, the ego-vehicle evolves its state estimate using its motion measurements; when the ego-vehicle has absolute positioning measurement of its own, it updates its state estimate according to the EKF.

Naïve Cooperative Localization Method (NCL):

Each ego-vehicle performs single vehicle localization as described above; besides, when the ego-vehicle receives the data from a neighboring vehicle, it treats the received data as new independent information and it updates its state estimate also using the EKF.

State Exchange based Cooperative Localization Method [Karam *et al.* 2006b] (SECL):
Each ego-vehicle maintains two state estimates. The first estimate is maintained as in single vehicle localization. When the ego-vehicle receives the data from neighboring vehicles, it forms the second estimate by using the EKF to fuse its first estimate and the received data. The second estimate (i.e. the fusion result of the data of the ego-vehicle and other vehicles) will neither be further used in the localization process of the ego-vehicle nor shared with other vehicles.

Cooperative Localization Method using the Split Covariance Intersection Filter (SCIFCL):
The proposed cooperative localization method as described in Section 3.4.

6.2.2 Simulation Scenario

A main scenario for comparative study is designed based on abstraction of real traffic scenarios and is illustrated in Figure 6.1: a chain of vehicles (for example, 8 vehicles) move on the same road in the same direction. Each vehicle is only able to observe its immediate neighbouring vehicles (as illustrated by the two-direction arrows), i.e. its immediate front vehicle and its immediate following vehicle.

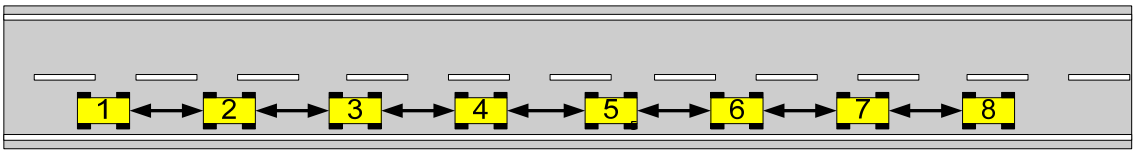


Figure 6.1 Simulation scenario: a chain of vehicles

The simulation conditions have been set according to the availability of the functionalities specified in Section 3.3 and can be changed. Experiments have been carried out under different simulation conditions; the condition which mainly affects the performance of each method is the absolute positioning error level. Therefore, without loss of generality, we tested the performance of the methods under different absolute positioning error level while fixing other simulation conditions.

The different absolute positioning error levels used for tests will be specified respectively in later sub-sections. Concerning other simulation conditions, we set the number of vehicles to be 8; the interval distance between neighboring vehicles to be 20 meters; the velocity of each vehicle to be 50 km/sec; the motion measurement standard errors to be 0.1 m/sec in velocity and 0.005 rad/sec in yawrate; the absolute positioning measurement period to be 1 second; the system period to be 0.1 second; the relative positioning standard errors to be 0.1 meter in relative position and 0.005 rad in relative orientation. We temporarily neglect in simulation the issue of communication delay, considering that the communication delay is usually short (no more than dozens of milliseconds) and that the errors caused by communication delay can be compensated by motion data as described in Section 3.4.2.

6.2.3 Homogeneous Systems: All Vehicles with the Same Absolute Positioning Ability

In this experiment, we let all the vehicles have the same absolute positioning ability; the absolute positioning standard error for each vehicle is set to be 5 meters. The simulation is carried out in the following way: at the first stage, each vehicle only uses the SL method until its own state estimate converges; then at the second stage, the SL method, the NCL method, the SECL method and the SCIFCL method are executed simultaneously and vehicle localization errors associated respectively with all these methods are collected for comparison.

The vehicle localization errors of one round of test are demonstrated in Figure 6.2 as an example. There are several sub-figures; a sub-figure displays the localization errors of a vehicle using the different localization methods in comparison. The localization errors associated with these methods are distinguished by different types of line with different colours. The vertical coordinates indicates the position error and the horizontal coordinates indicates the time sequence. As we can see, the estimate obtained by the NCL method severely diverges, which shows that careless handling of the inter-estimates correlation in cooperative localization will easily incur the over-convergence problem.

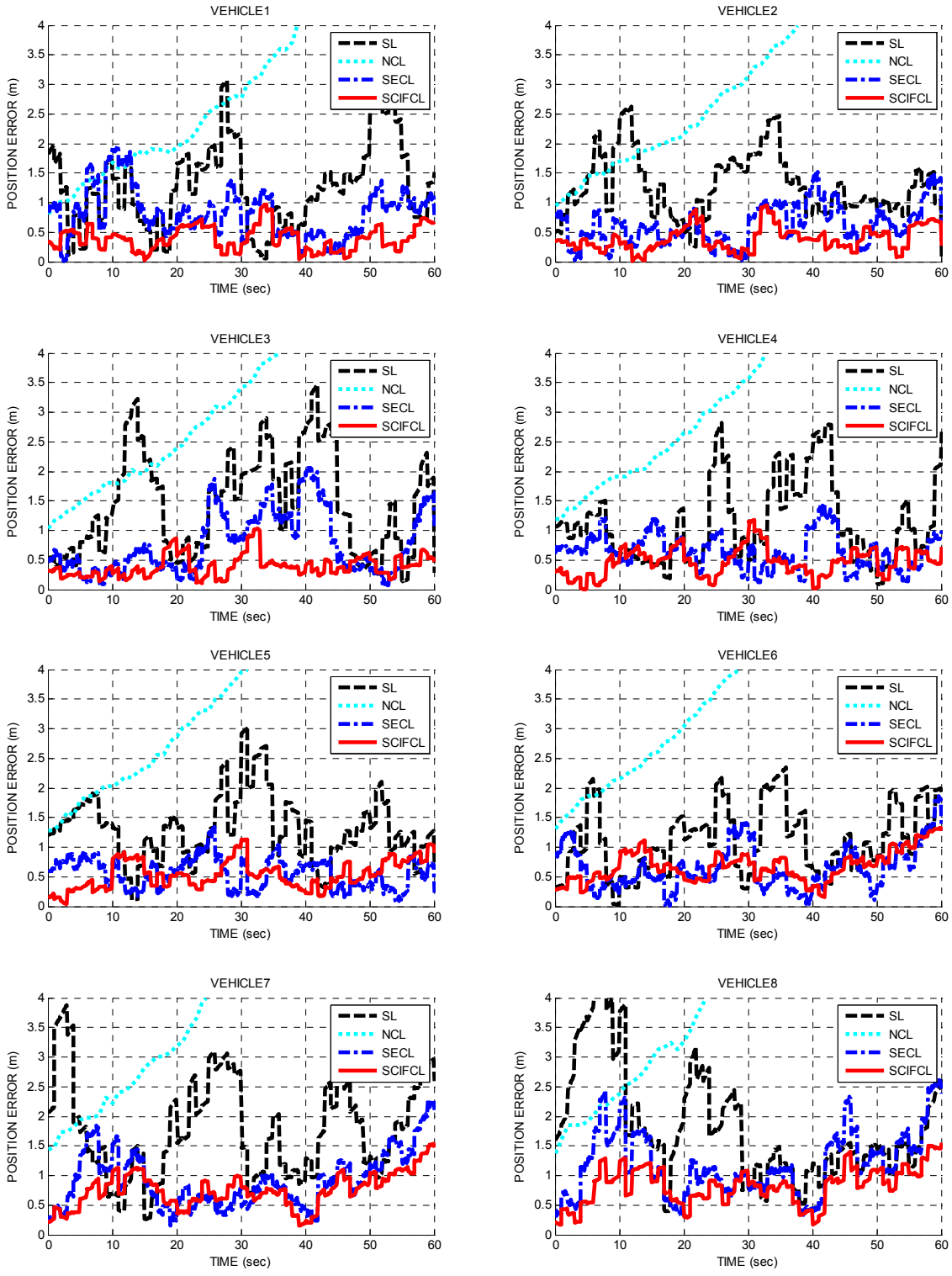


Figure 6.2 Performance of the SL method, the NCL method, the SECL method and the SCIFCL method (homogeneous absolute positioning ability)

On the other hand, both the SECL method and the SCIFCL method achieve better performance (in terms of localization accuracy) than the SL method on the whole, which shows that cooperative localization methods, if well designed, can considerably improve the performance of vehicle localization.

The result shown in Figure 6.2 gives an intuitive comparison among the performance of the four methods. Furthermore, a large number of tests (totally fifty rounds of test) have been carried out to have a quantitative comparison among these methods. In every round of test, the RMS (Root Mean Square) of the position errors of all the vehicles, associated with each of the SL method, the SECL method and the SCIFCL method (the NCL method is excluded for comparison because it usually leads to severely diverged result), is computed. The results are demonstrated in Figure 6.3, where the vertical coordinates indicates the computed RMS and the horizontal coordinates indicates the indices of the round of test.

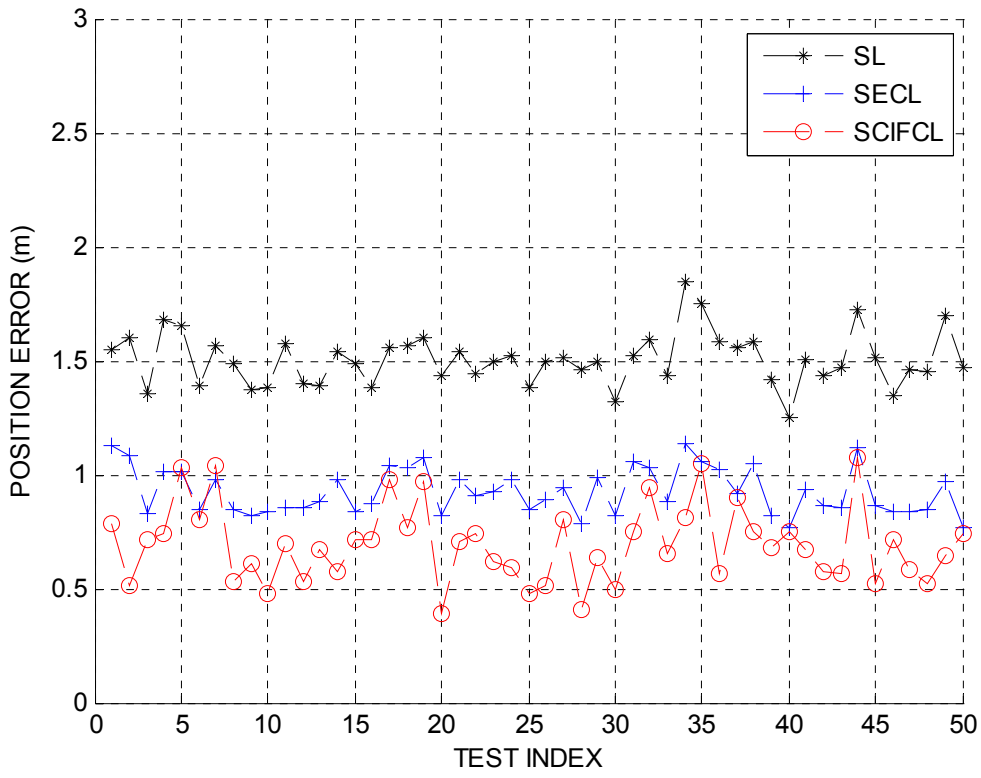


Figure 6.3 RMS of the localization error associated with the SL method, the SECL method and the SCIFCL method (homogeneous absolute positioning ability)

As we can see in Figure 6.3, the SECL method and the SCIFCL method always outperform the SL method, which shows the advantage of cooperative localization over single vehicle localization. The SCIFCL method also outperforms slightly the SECL method (the RMS errors of the 50 rounds of test for the SCIFCL and SECL method are respectively 0.71 meter and 0.92 meter).

6.2.4 Heterogeneous Systems: One Vehicle with High-Accuracy Absolute Positioning Ability

In this experiment, we let all the vehicles have the same low-accuracy absolute positioning ability, except the first vehicle (the left-most vehicle in Figure 6.1) which has high-accuracy absolute position ability. More specifically, the absolute position standard error for the first vehicle is set to be 0.1 meter, whereas that for other vehicles (the second vehicle to the eighth vehicle) is set to be 15 meters. The simulation is carried out in the same way as in the last experiment for *Homogeneous Systems*.

The vehicle localization errors of one round of test are demonstrated in Figure 6.4 as an example. As in Figure 6.2, each sub-figure displays the localization errors associated with one vehicle; the vertical coordinates indicates the position error and the horizontal coordinates indicates the time sequence. The estimate obtained by the NCL method severely diverges; especially, we can pay attention to the first vehicle. In spite of the high-accuracy absolute positioning measurements the first vehicle has, yet its state estimate diverges far and far away from the ground-truth.

The SECL method still achieves better performance than the SL method on the whole. For the second vehicle, its localization result is largely improved by the high-accuracy localization result of the first vehicle through cooperative localization. However, in the SECL method, an estimate obtained by fusing the data of different vehicles is not allowed to be further used, which limits the improvements that the SECL method can bring to the localization of the third to the eighth vehicles.

The performance of the SCIFCL method is distinguished compared to that of the SECL method and the SL method; the localization results of all the vehicles are largely improved, especially for the third to the eighth vehicles. The reason can be understood as follows: through cooperative localization using the SCIFCL method, the high-accuracy localization result of the first vehicle can improve the localization result of the second vehicle; the improved localization result of the second vehicle can further improve the localization result of the third vehicle and so on. This is like the “good” localization result originated from the first vehicle can be propagated to the second

vehicle, then to the third vehicle, then to the next vehicle until the eighth vehicle. Although the third to the eighth vehicles are not in the immediate neighborhood of the first vehicle, they can still indirectly benefit from the “good” data of the first vehicle.

It is worthy reminding that NO monitoring and controlling of data flow is performed in the SCIFCL method. We do not deliberately control the data to flow successively from the first vehicle to the eighth vehicle; the localization results of two neighbouring vehicles are mutually influencing. The SCIFCL method enables good localization results to be naturally spread within a vehicle network in connection while always keeping a reasonable confidence for the state estimate of each vehicle.

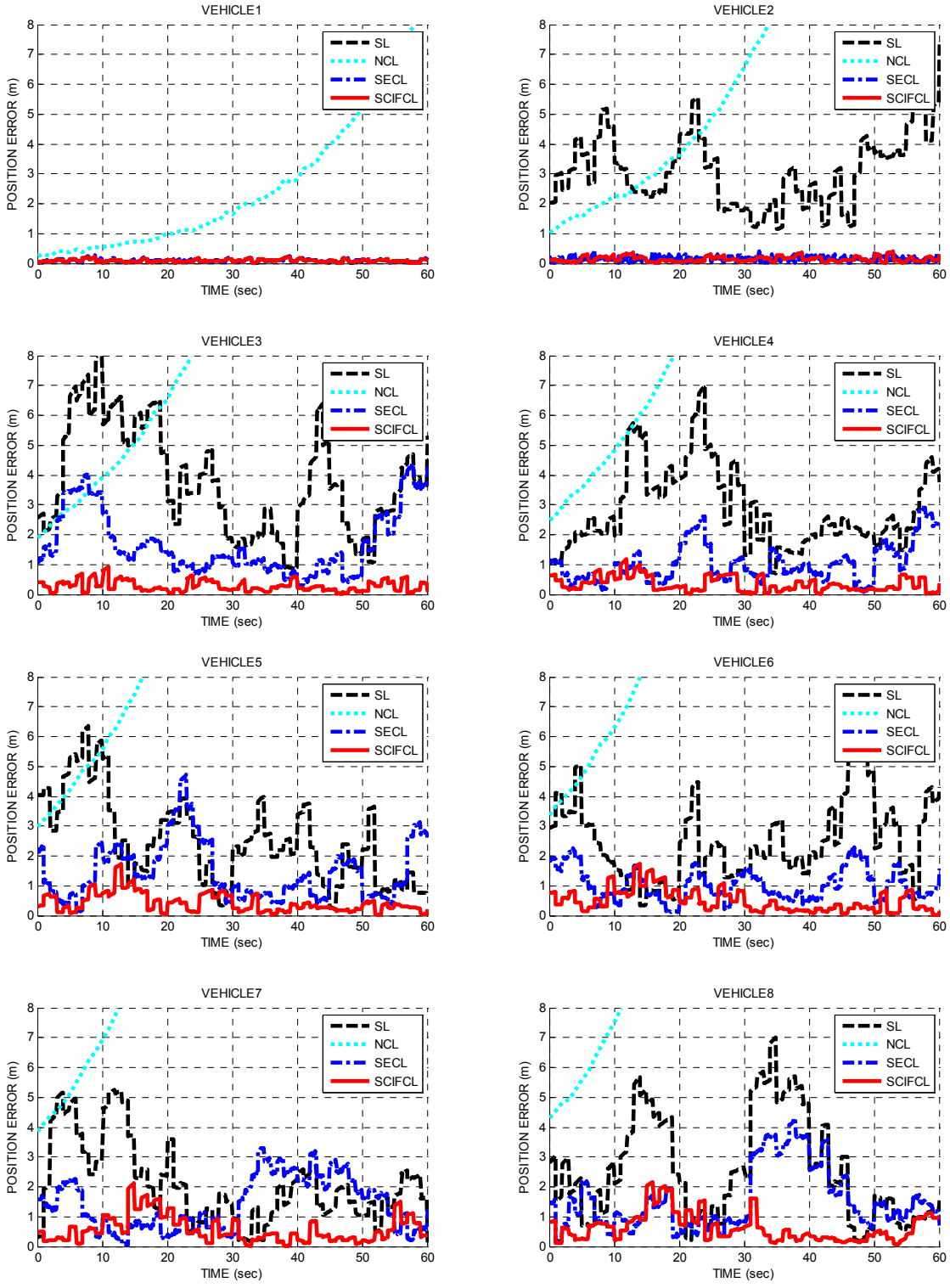


Figure 6.4 Performance of the SL method, the NCL method, the SECL method and the SCIFCL method (heterogeneous absolute positioning ability)

Fifty rounds of tests have been carried out to have a quantitative comparison among the SL method, the SECL method and the SCIFCL method. As in the last experiment for *Homogeneous Systems*, in every round of test, the vehicles position error RMS of each method is computed. The results are demonstrated in Figure 6.5, where the vertical coordinates indicates the computed RMS and the horizontal coordinates indicates the indices of the round of test.

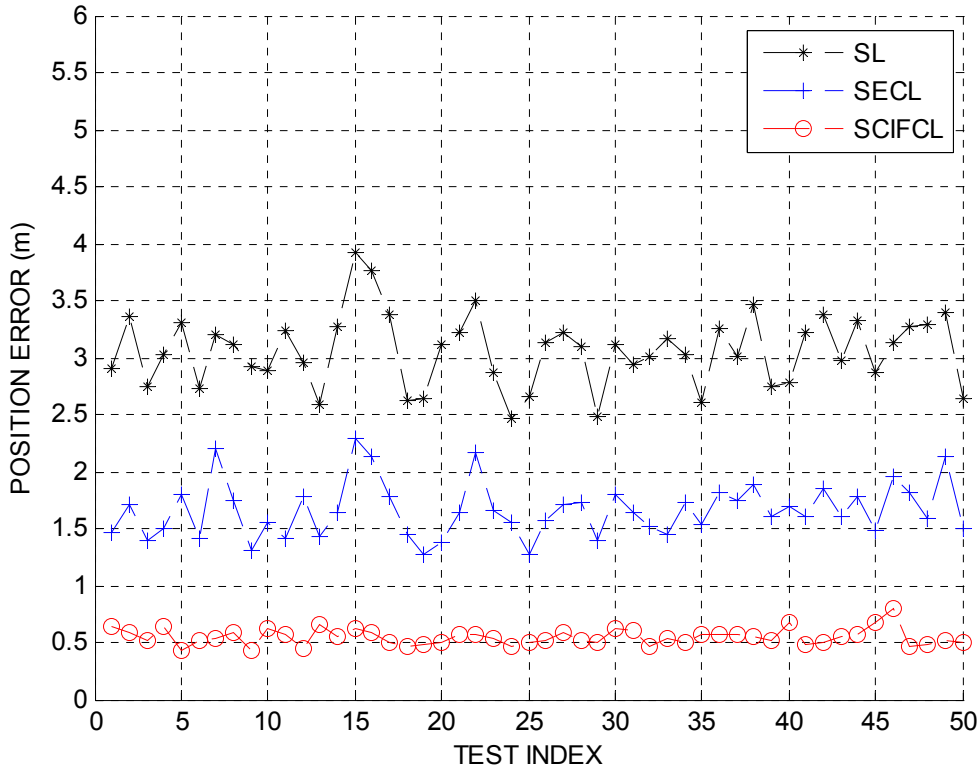


Figure 6.5 RMS of the localization error associated with the SL method, the SECL method and the SCIFCL method (heterogeneous absolute positioning ability)

As we can see in Figure 6.5, the SCIFCL method always yields apparent performance improvement over the SL method and the SECL method, which demonstrates the effectiveness and the advantage of the proposed cooperative localization architecture using split covariance intersection filter.

6.2.5 Discussion

Two kinds of experiments have been described in previous sub-sections. The experiment for homogeneous systems is intended to demonstrate the statistical advantage of cooperative localization using the SCIFCL method. In reality, each vehicle usually has few neighboring vehicles to cooperate with (for example, just the front one and the following one); as a consequence, this statistical advantage might be quite limited for intelligent vehicles with homogeneous absolute positioning ability in practical applications.

On the other hand, cooperative localization is more valuable and practical for intelligent vehicles with heterogeneous absolute positioning ability, as demonstrated in the experiment for heterogeneous systems. A prominent advantage of the SCIFCL method is that it enables good localization results to be naturally spread within a vehicle network in connection while always keeping a reasonable confidence for the state estimate of each vehicle.

The significance of cooperative localization demonstrated by the experiment for heterogeneous systems can be interpreted as follows: Suppose there are several vehicles in neighborhood; each vehicle might randomly loose their accurate absolute positioning ability. During cooperative localization, if only one vehicle can possess accurate absolute positioning ability, then other vehicles can also obtain rather accurate localization results. From statistical viewpoint, at a certain time, although some vehicles might temporarily loose their accurate absolute positioning ability, it is very UNLIKELY that all the vehicles loose their accurate absolute positioning ability. We can do a simple statistical calculation: Suppose there are N_v=8 vehicles and each vehicle has accurate absolute positioning ability during only half time (p_{single}=50%), then the percentage of time when all these vehicles can maintain rather accurate localization results would be as high as: $p_{group}=1-(1-p_{single})^{Nv}$=99.6%, i.e. almost all the time.

6.3 Field Tests on Cooperative Localization

6.3.1 Experimental Conditions

Real data were logged in INRIA campus (as shown in Figure 6.6), based on two CyCab vehicle platforms developed by INRIA-IMARA team. Each CyCab vehicle is equipped with a RTK-GPS, an IBEO laser scanner, and odometer sensors (including steering encoder). A RTK-GPS can achieve centimeter-level positioning accuracy. The vehicle trajectory registered by a RTK-GPS (interpolated with corrected motion data) is used as

the ground-truth for results comparison; an example is illustrated by the red line in Figure 6.6. The time of the two vehicle systems are related to the GPS universal time.

Real data experiments were carried out on the cooperative localization methods (the SL method, the NCL method, the SECL method and the proposed SCIFCL method), similar to the simulation experiments demonstrated in Section 6.2. In the real data tests, the RTK-GPS outputs were deliberately degraded with random errors and were used as the absolute positioning measurements. The motion data (speed and yawrate) were provided (or indirectly computed) by the odometers and were corrected by laser scanner based local SLAM. The vehicle-to-vehicle relative poses (relative positioning) were estimated using the method described in Section 3.5. As in simulation experiments, we have also carried out two sets of experiments: one for homogeneous systems, the other for heterogeneous systems.

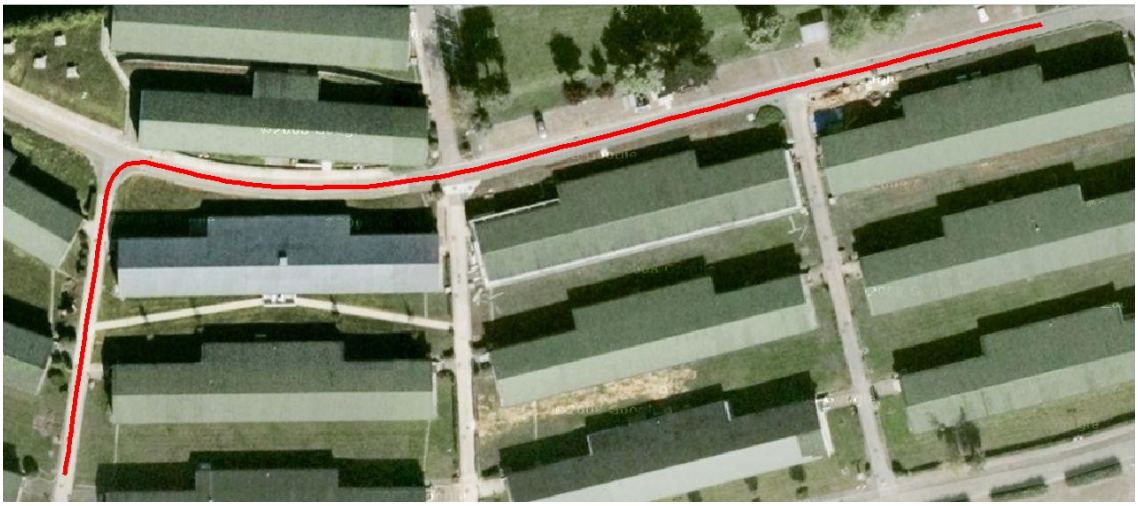


Figure 6.6 Experimentation field and the ground-truth of one vehicle trajectory

6.3.2 Homogeneous Systems: All Vehicles with the Same Absolute Positioning Ability

As in the simulation experiments for homogeneous systems (Section 6.2.3), we let all the vehicles have the same absolute positioning ability; the standard error used to deliberately degrade the RTK-GPS outputs is set to be 5 meters. The experiments were carried out in the same way as in the simulation experiments, i.e. the SL method, the NCL method, the SECL method and the SCIFCL method are executed simultaneously on the same data and vehicle position errors associated respectively with all these methods are collected for comparison.

The vehicle position errors of one round of test are demonstrated in Figure 6.7. We can see that the estimate obtained by the NCL method severely diverges, same to the result in simulation tests.

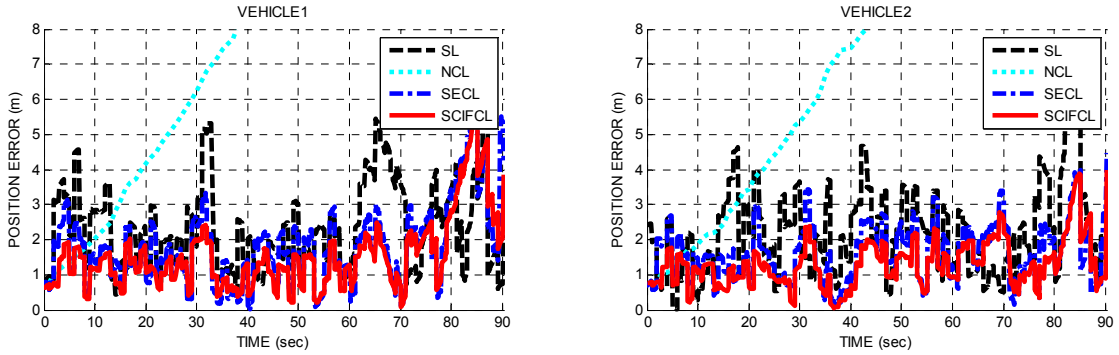


Figure 6.7 Performance of the SL method, the NCL method, the SECL method and the SCIFCL method (homogeneous absolute positioning ability)

Fifty rounds of tests have been carried out to have a quantitative comparison among the SL method, the SECL method and the SCIFCL method. In every round of test, the vehicles position error RMS of each method is computed. The results are demonstrated in Figure 6.8, where the vertical coordinates indicates the computed RMS and the horizontal coordinates indicates the indices of the round of test. As we can see, there is moderate statistical advantage of cooperative localization over single vehicle localization; the limited statistical advantage is due to the few vehicle number (only two).

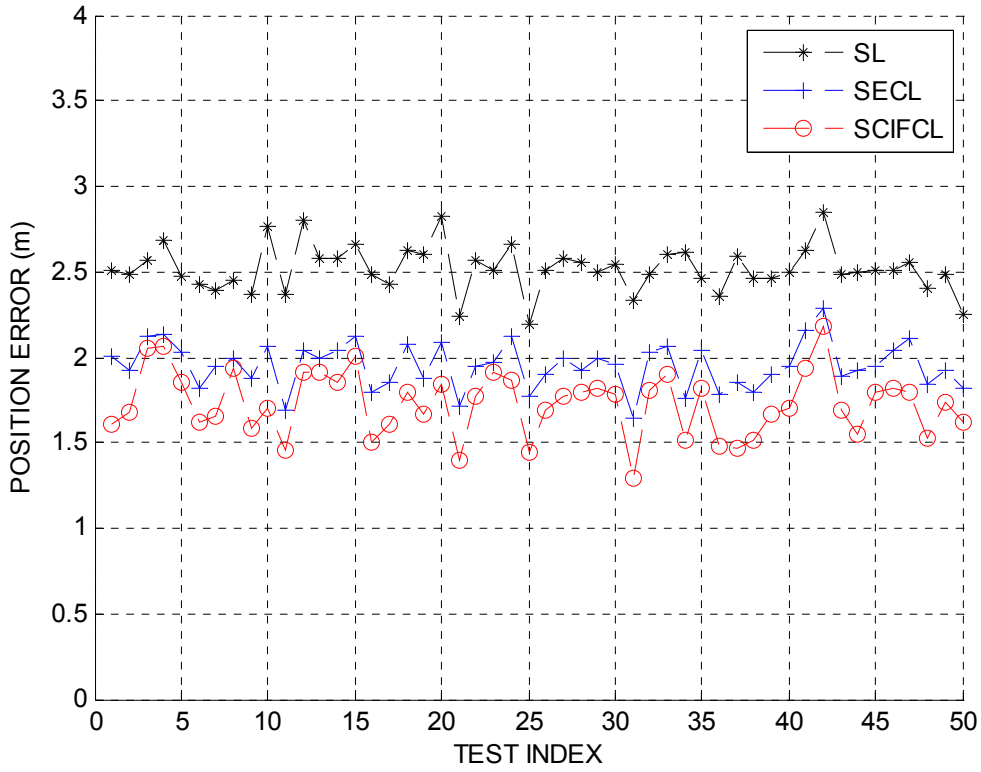


Figure 6.8 RMS of the localization error associated with the SL method, the SECL method and the SCIFCL method (homogeneous absolute positioning ability)

Certain special cases can highlight the advantage of cooperative localization; For example, when the bias errors (see the discussion in this sub-section later) of different GPS happen to counteract each other. We have carried out a test to demonstrate the apparent advantage of cooperative localization in such kind of special case. We degraded the GPS output of one vehicle with an extra bias error of (7, -8) meters (in the vehicle moving plane); we degraded that of the other vehicle with an extra bias error of (-6, 9) meters; the localization results of one round of test is illustrated in Figure 6.9. For the first vehicle, the root mean square (RMS) of its self-localization error is 10.24m and the RMS of its cooperative localization error is 2.98m. For vehicle 2, the RMS of its self-localization error is 10.64m and the RMS of its cooperative localization error is 2.93m. In such kind of special case, much better localization results are achieved via cooperative localization.

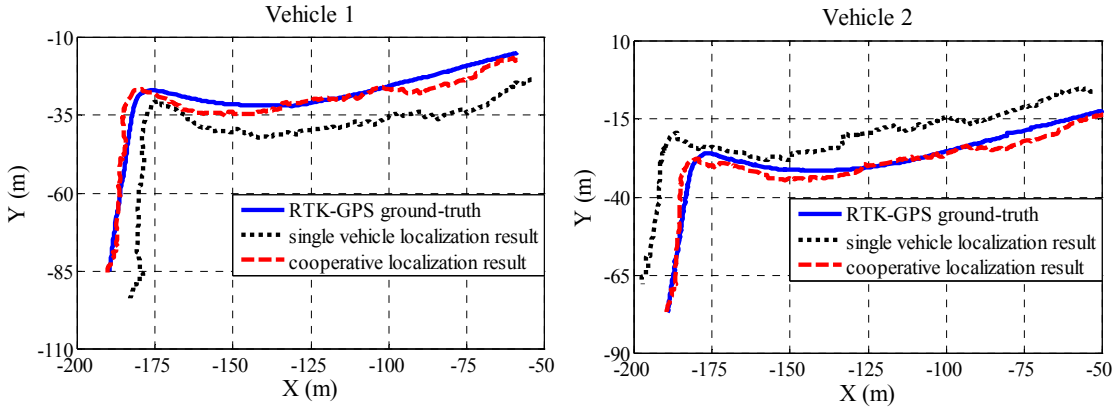


Figure 6.9 Localization results at the existence of counteracting GPS bias errors

6.3.3 Heterogeneous Systems: One Vehicle with High-Accuracy Absolute Positioning Ability

As in the simulation experiments for heterogeneous systems (Section 6.2.4), we let one vehicle has comparatively high-accuracy absolute positioning ability; the standard error used to degrade its GPS output is set to be only 1.0 meter. The standard error used to degrade the GPS output of the other vehicle is set to be 15 meters. The experiments were carried out in the same way as for previous experiments demonstrated.

The vehicle position errors of one round of test are demonstrated in Figure 6.10; The quantitative results of fifty rounds of tests are demonstrated in Figure 6.11. Thanks to cooperative localization which enables the second vehicle to take advantage of the data of the first vehicle, the localization errors of the second vehicle are largely reduced. Since we only had two vehicles for experimentation, we could not arrange certain experimental scenarios where some vehicles can not directly cooperate with the vehicle with better absolute positioning ability. Therefore, we can not demonstrate the apparent advantage of the proposed SCIFCL method over the SECL method, as demonstrated in the simulation experiments for heterogeneous systems (Section 6.2.4).

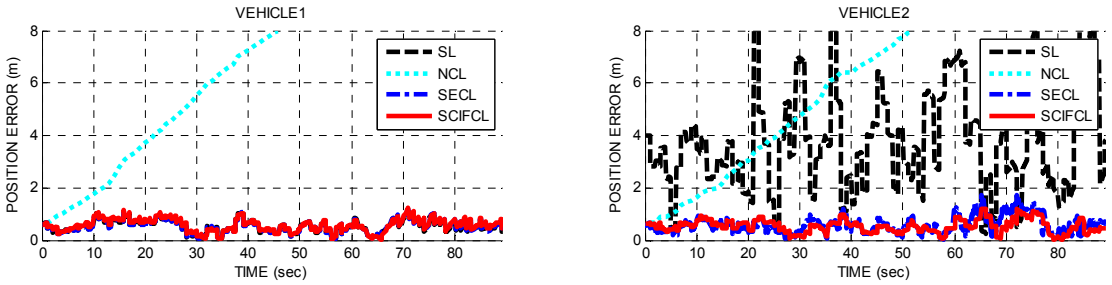


Figure 6.10 Performance of the SL method, the NCL method, the SECL method and the SCIFCL method (heterogeneous absolute positioning ability)

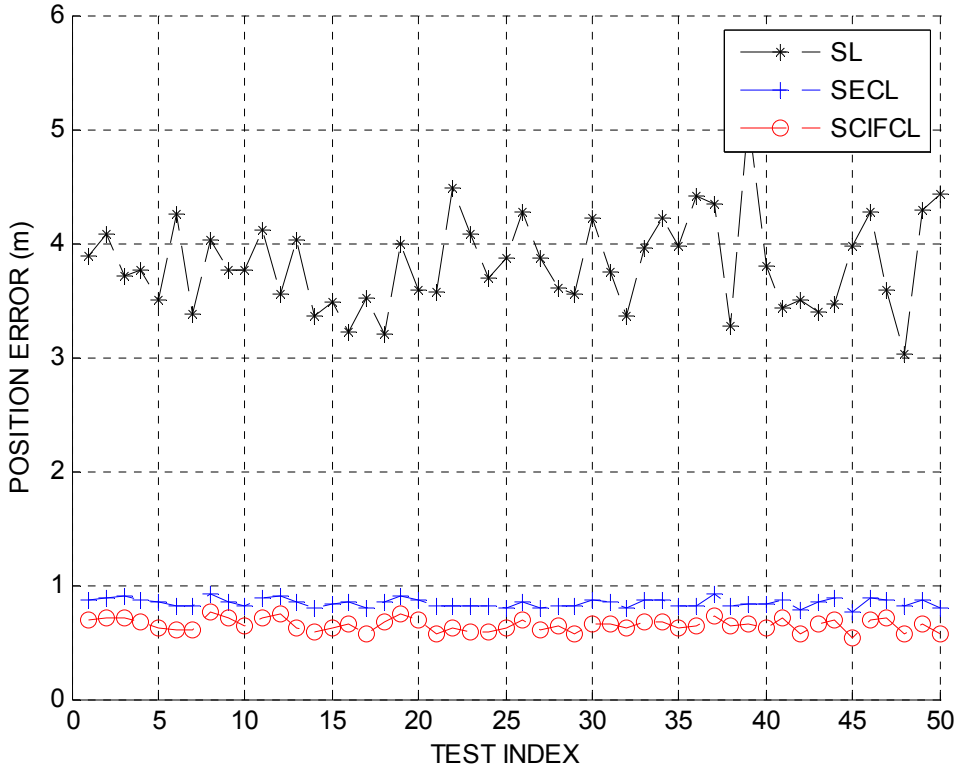


Figure 6.11 RMS of the localization error associated with the SL method, the SECL method and the SCIFCL method (heterogeneous absolute positioning ability)

When GPS bias errors exist, the advantage of cooperative localization for heterogeneous systems is also apparent. We degraded the GPS output of one vehicle with Gaussian noise of standard error 7 meters and an extra bias error of (-6, 9) meters. The standard error used to degrade the GPS output of the other vehicle is still set to be

only 1.0 meter. The estimated vehicle trajectories using different methods are illustrated in Figure 6.12. As we can see, the estimated vehicle trajectory (the second vehicle) using the SCIFCL method is much closer to the ground-truth.

Figure 6.12 Estimated vehicle trajectories using the CL method and the SCIFCL method at the existence of GPS bias errors: (top) the first vehicle, with better absolute positioning ability; (bottom) the second vehicle, with low-accuracy absolute positioning accuracy

6.3.4 Discussion

According to some research works [Laneurit *et al.* 2005], it seems that GPS errors can be modeled by a white Gaussian noise combined with a slowly-changing bias vector. Since in above experiments (synthetic data or real data), the absolute positioning

measurement error is assumed to follow zero-mean Gaussian distribution, a question arises naturally: what is the performance of the proposed cooperative localization method?

In fact, when GPS accuracy is low, the GPS errors might had better be modeled as a combination of a slowly-changing bias vector and a white noise as in [Laneurit *et al.* 2005]. On the other hand, for high-accuracy GPS (with sub-meter level precision), according to our experience, the GPS error can be fairly modeled by a zero-mean Gaussian. Therefore, the proposed cooperative localization method would be expected to bring benefits for multiple vehicles at least one of which has rather accurate absolute positioning ability, as demonstrated in the experiments for *heterogeneous systems*. As we have discussed in Section 6.2.5, cooperative localization is more valuable and practical for intelligent vehicles with heterogeneous absolute positioning ability.

6.4 Cooperative Local Mapping and Moving Objects Detection

6.4.1 Experimental Conditions

Real data experiments were carried out in INRIA campus, based on two CyCab vehicle platforms developed by INRIA-IMARA team. Each CyCab vehicle is equipped with a RTK-GPS, an IBEO laser scanner, and odometer sensors (including steering encoder).

A RTK-GPS can achieve centimeter-level positioning accuracy; however, we do not assume in our method that an intelligent vehicle should possess such high-quality configuration. In reality, an intelligent vehicle might be equipped not with a RTK-GPS but with a normal low-cost GPS, out of economical considerations. Even the availability of a RTK-GPS can not always guarantee centimeter-level positioning accuracy; the RTK-GPS positioning accuracy might be degraded to ten meters level due to signal blocking. Therefore, in order to simulate intelligent vehicles with low-cost GPS, we deliberately degraded the GPS outputs of both vehicles with random errors. The time of the two vehicle systems are related to the GPS universal time (a low-cost GPS can also obtain accurate GPS universal time).

More specifically, the random error used to degrade a RTK-GPS is comprised of a bias error of 10 meters and a white noise of standard error 7 meters; the bias error vector is initialized randomly and assumed to be temporarily constant for each vehicle in a round of test (Note: according to some research works [Laneurit *et al.* 2005], GPS errors could be modeled by a combination of a white Gaussian noise and a slowly-changing bias vector).

6.4.2 Occupancy Grid Maps Merging: Ground-Truth

Both CyCab vehicles perform local SLAM simultaneously; at each time, a pair of local occupancy grid maps from the two vehicles can be obtained—For maps merging itself, the generation of the two local maps do not need to be well synchronized or time-stamped accurately, because a map reference (at any time, build by any vehicle) is always fixed with the environment. This is different from vehicle references which might move and whose motion should be well time-stamped—for each pair of local occupancy grid maps, we do not know the ground-truth of their correct alignment. Although we can measure the ground-truth of vehicles positions using RTK-GPS, we do not know the ground-truth of the relative pose between each vehicle and its local environment. The relative pose between the two local maps **A** and **B** can only be inferred indirectly (the meaning of the denotations is referred to Section 3.5):

$$\mathbf{p}_{\mathrm{BA(RTK\text{-}GPS)}} = \mathbf{p}_{\mathrm{LA}} \oplus \mathbf{p}_{\mathrm{vBA(RTK\text{-}GPS)}} \oplus inv(\mathbf{p}_{\mathrm{LB}}) \tag{6-1}$$

The $\mathbf{p}_{\mathrm{vBA(RTK\text{-}GPS)}}$ can be computed from the RTK-GPS based estimates; the $\mathbf{p}_{\mathrm{LA}}$ and $\mathbf{p}_{\mathrm{LB}}$ can only be obtained from the local SLAM results. The estimated $\mathbf{p}_{\mathrm{vBA(RTK\text{-}GPS)}}$, $\mathbf{p}_{\mathrm{LA}}$ and $\mathbf{p}_{\mathrm{LB}}$ still have certain level of errors, especially the orientation error. After the compounding operation in (6-1), the errors in them will be propagated and amplified. As a result, if we align the two local maps using the $\mathbf{p}_{\mathrm{BA(RTK\text{-}GPS)}}$ computed in (6-1), there will be slight inconsistency between the aligned maps, as illustrated in the bottom-right sub-figure in Figure 6.13.

In order to determine the correct alignment (the ground-truth), we carry out a dense searching in a small range around of this initial alignment $\mathbf{p}_{\mathrm{BA(RTK\text{-}GPS)}}$ and choose the one with highest fitness value as the ground-truth, the merging result using the correct alignment is demonstrated in the top-right sub-figure in Figure 6.13.

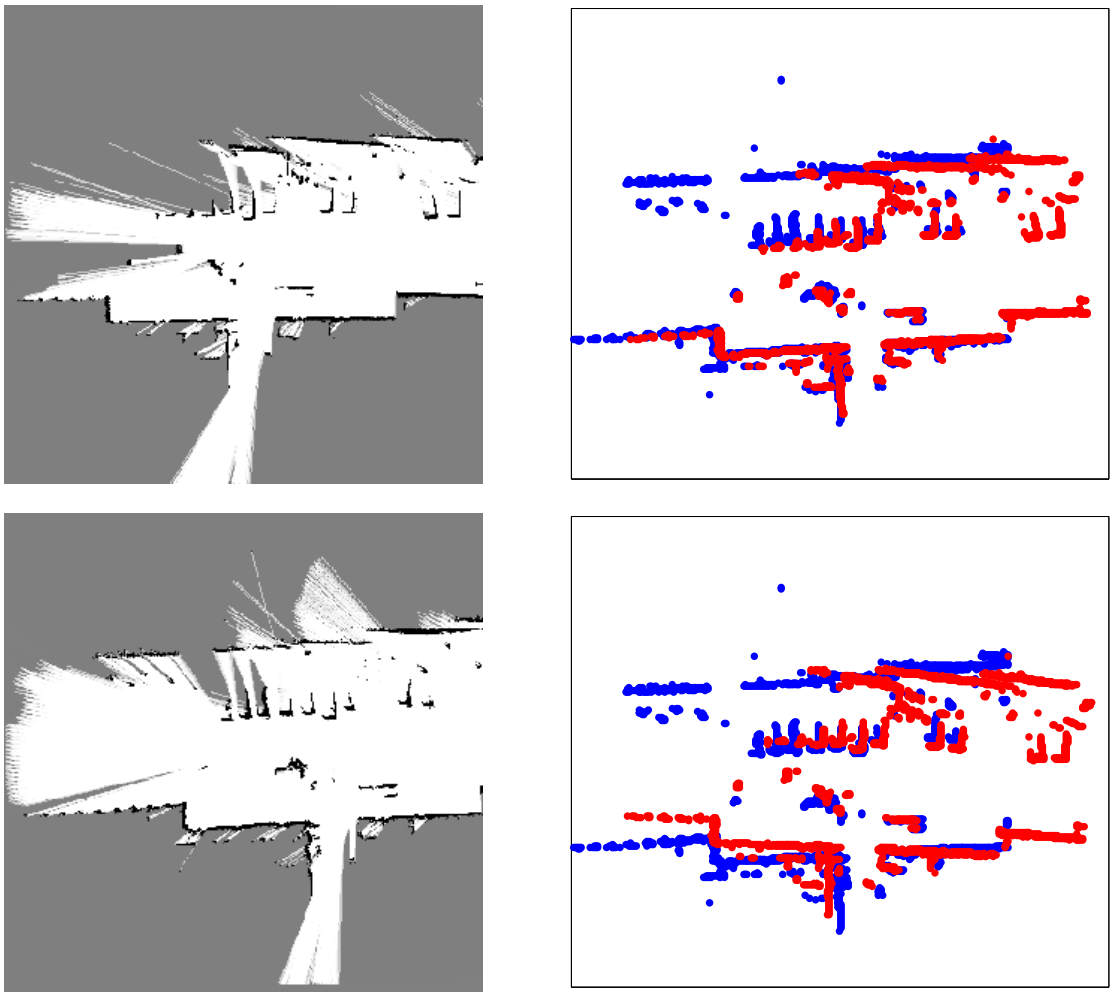


Figure 6.13 Ground-truth of local maps alignment: (Left) the two local occupancy grid maps; (Top-Right) the ground-truth; (Bottom-Right) Slight inconsistency

6.4.3 Occupancy Grid Maps Merging: Experiment I

The occupancy grid maps merging method introduced in Section 4.3 is tested on totally 1155 pairs of local occupancy grid maps: for each pair, we randomly generated an initial maps alignment in a deliberately exaggerated error range around the ground-truth; the error range is $\pm$30 meters in position (both horizontally and vertically) and $\pm$30 degrees in orientation (see Section 4.3.3); then we used the proposed occupancy grid maps merging method to merge the pair of local maps.

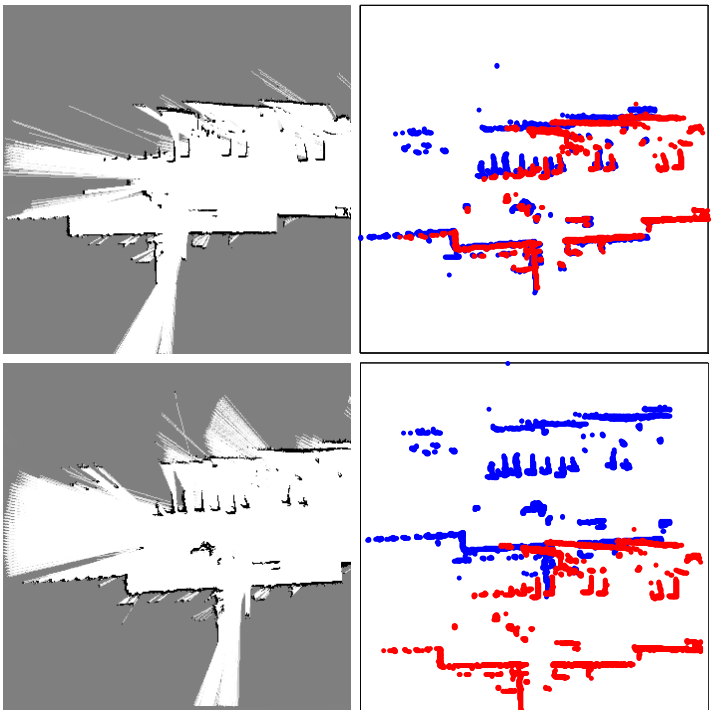


Figure 6.14 Occupancy grid maps merging effect

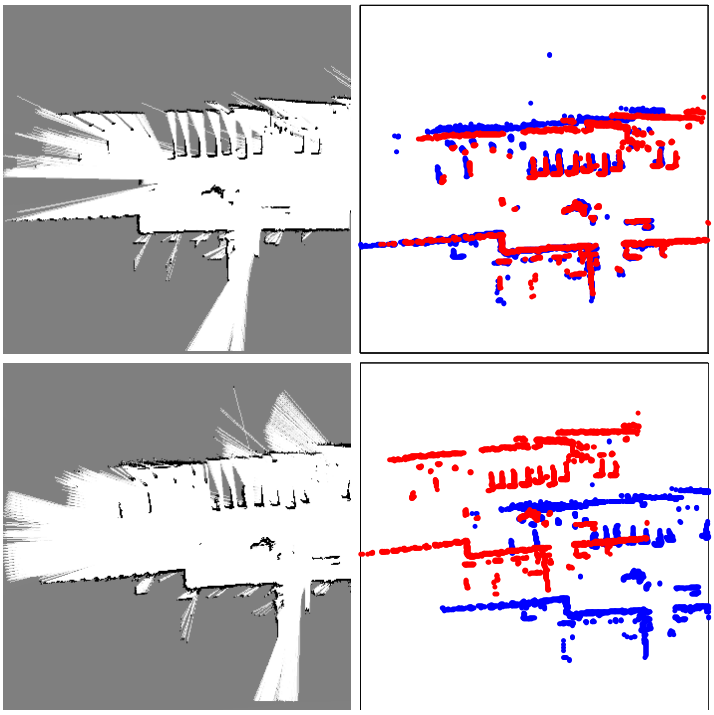


Figure 6.15 Occupancy grid maps merging effect

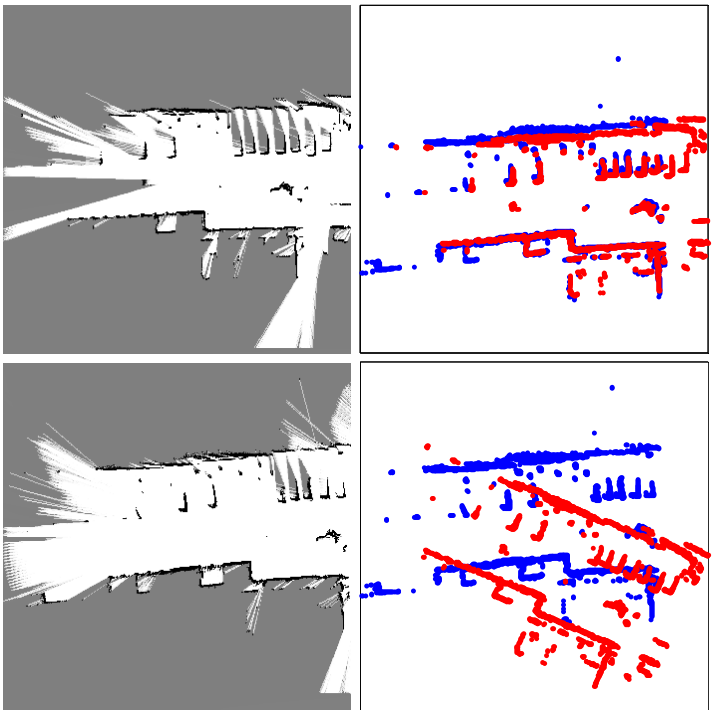


Figure 6.16 Occupancy grid maps merging effect

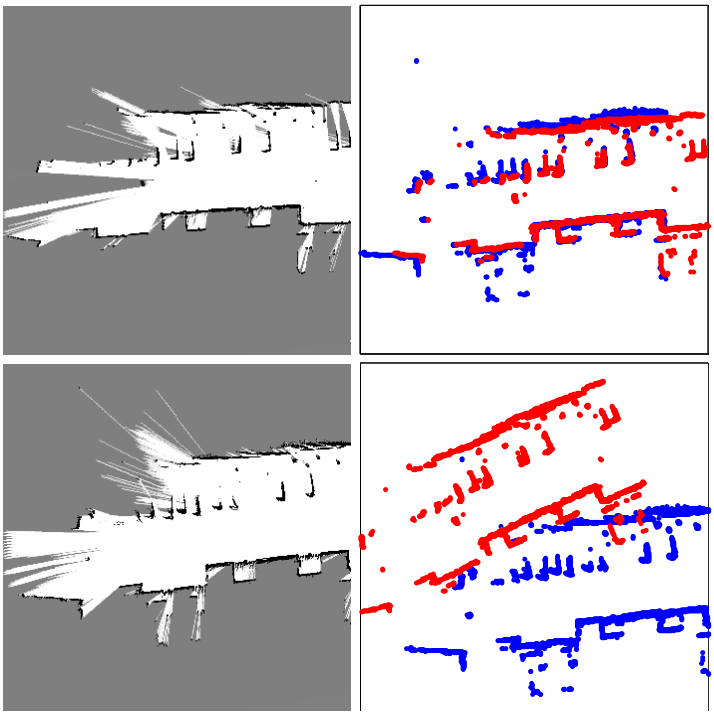


Figure 6.17 Occupancy grid maps merging effect

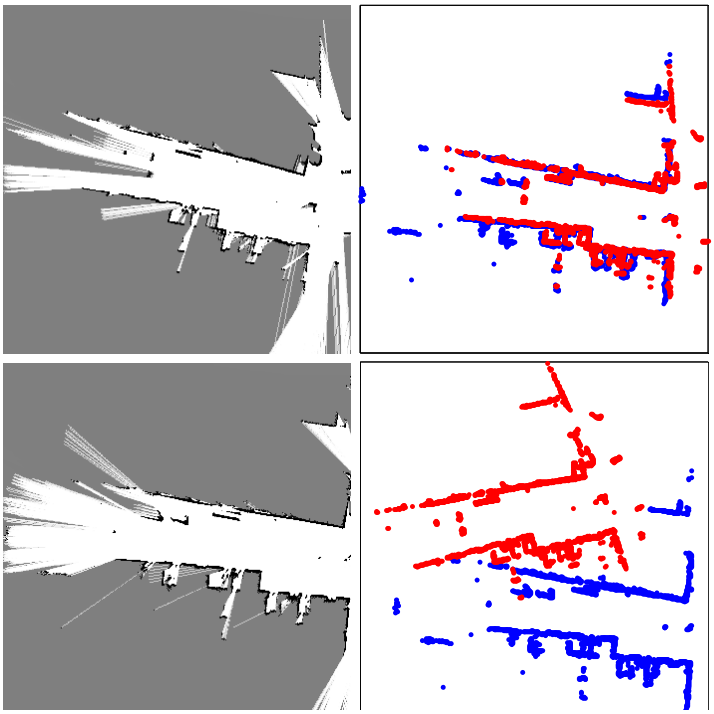


Figure 6.18 Occupancy grid maps merging effect

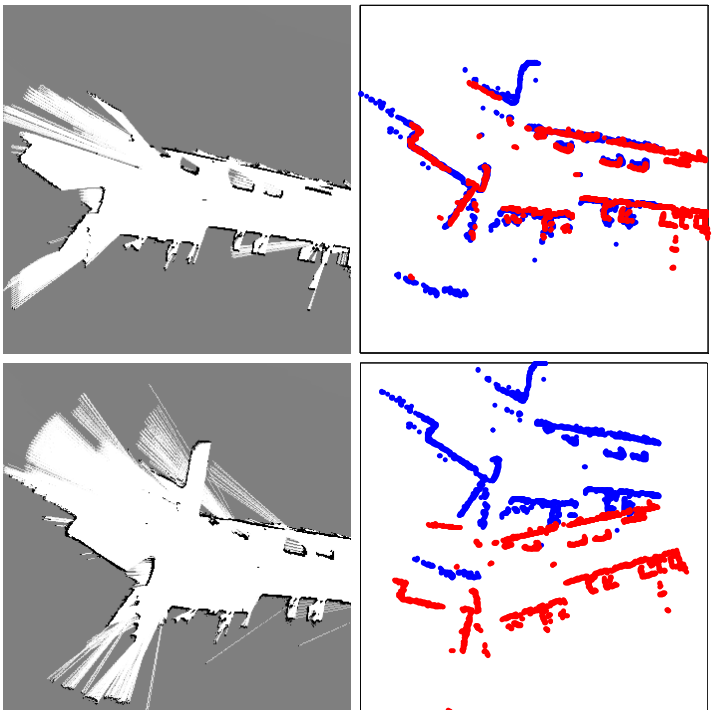


Figure 6.19 Occupancy grid maps merging effect

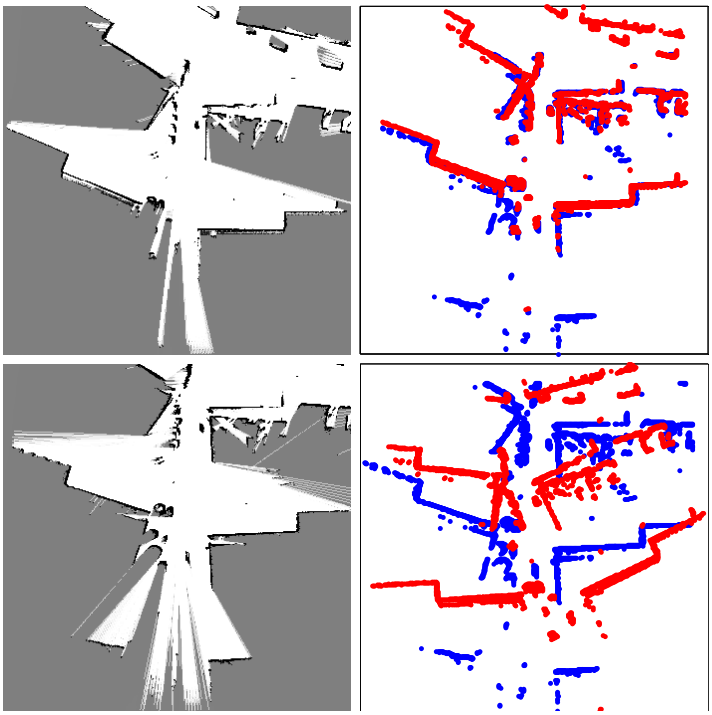


Figure 6.20 Occupancy grid maps merging effect

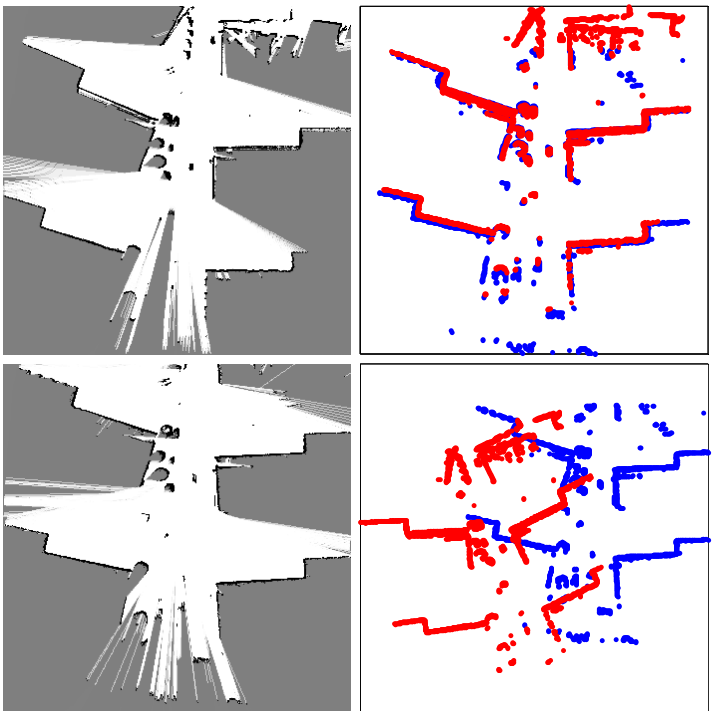


Figure 6.21 Occupancy grid maps merging effect

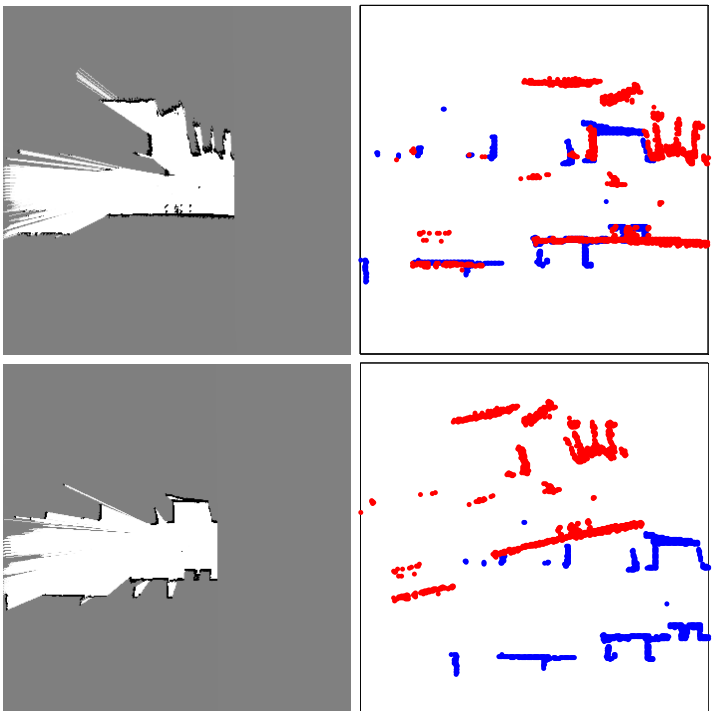


Figure 6.22 Occupancy grid maps merging effect

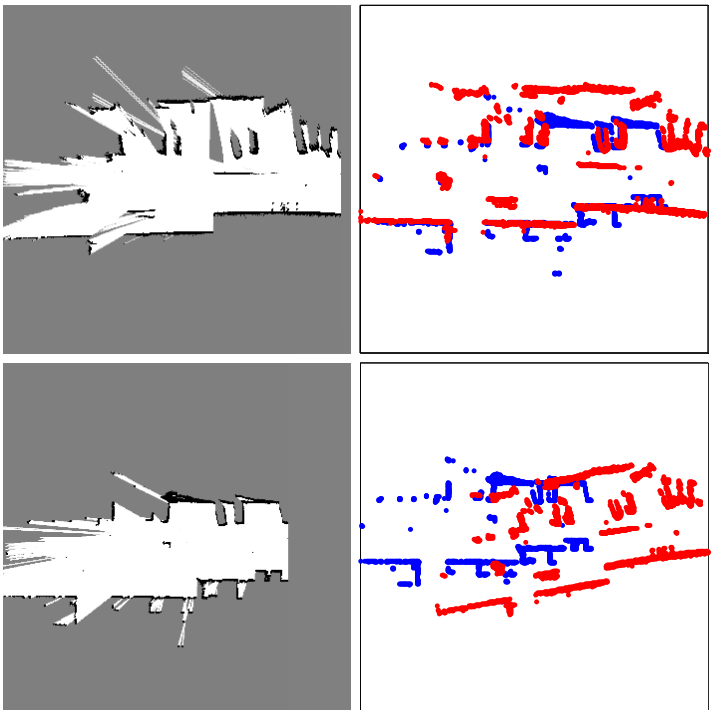


Figure 6.23 Occupancy grid maps merging effect

The genetic evolution based optimization introduced in Section 4.3.3 is executed until certain convergence criterion is satisfied, i.e. the optimization output is within a certain error range around the ground-truth. The error range is set to be 20×20 cm^2 (the occupancy grid cell size) in position and half a degree in orientation. The population size in the genetic algorithm is controlled at a level such that one round of evolution can be performed within one system period (0.1 second); in our implementation, it is set to be 1000.

Some examples of occupancy grid maps merging effect are demonstrated in Figures from Figure 6.14 to Figure 6.23. In each figure, the left two sub-figures display a pair of local occupancy grid maps built by the two CyCab vehicles; the bottom-right sub-figure displays the erroneous initial maps alignment; the top-right sub-figure displays the correct alignment after they are merged.

For the test on each pair of maps, we compute the *convergence evolution number*, i.e. the number of genetic evolutions needed for the optimization to converge. The histogram of the convergence evolution number for all the pairs under test is demonstrated in Figure 6.24. As we can see, the optimization process normally converges within ten times of evolution.

The average convergence evolution number is 5.46. We can make a rough analysis for the computational complexity of the genetic evolution based optimization method and that of an exhaustive searching based optimization method. We treat the computation of the fitness value of one individual as one operation. Each round of genetic evolution takes about 1000~2000 operations; let it be 1500 operations. The average number of operations needed for optimization convergence is about $5.46 \times 1500 \approx 8000$. For the exhaustive searching, in order to guarantee a searching accuracy of 20×20 cm^2 in position and 0.5 degree in orientation within an initial error range of 30×30 m^2 in position and 30 degrees in orientation, we averagely need $(30 \times 30 \times 30)/(0.2 \times 0.2 \times 0.5)$ =1350000 operations. Therefore, the genetic evolution based optimization is more efficient than the exhaustive searching based optimization by a factor of 1350000/8000=168.

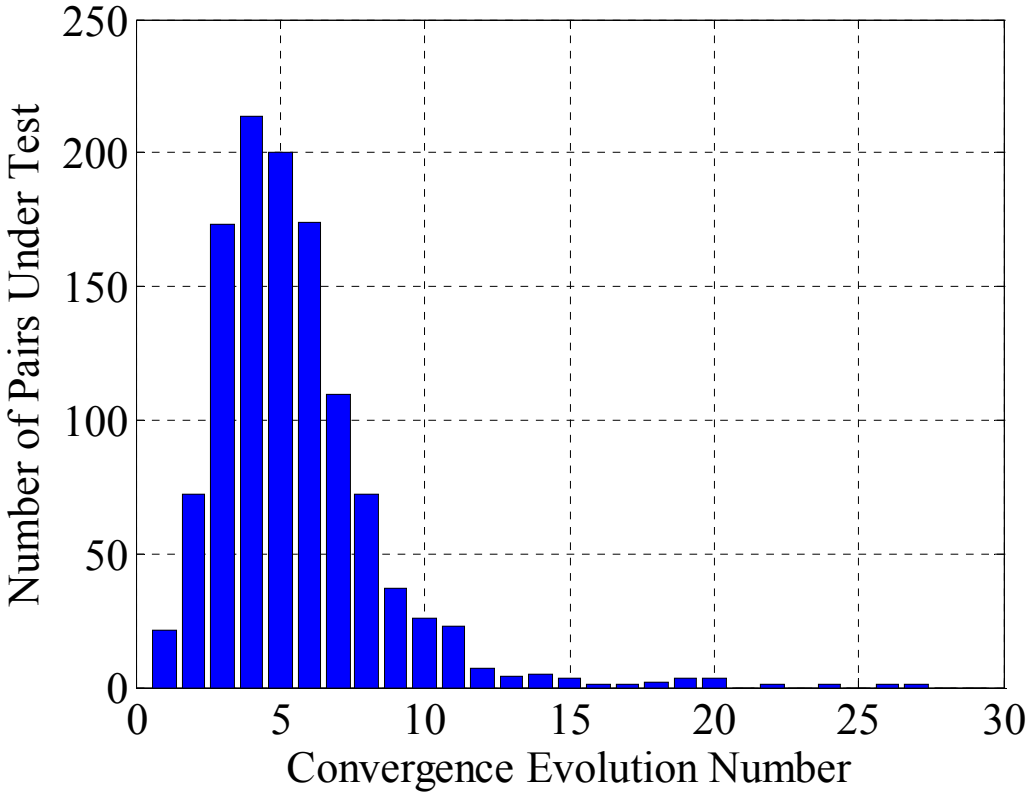


Figure 6.24 Histogram of the convergence evolution number

6.4.4 Occupancy Grid Maps Merging: Experiment II

In this experiment, we demonstrate how the proposed occupancy grid maps merging method can recover the merging result from a *kidnapping* situation. Given a pair of local occupancy grid maps to-be-merged, as illustrated in Figure 6.25; first, they are merged using the proposed occupancy grid maps merging method. The process of genetic evolution is illustrated in Figure 6.26; in each sub-figure, the black points represent the positions of the individuals, each of which describes a tentative alignment of the two maps. The maps are aligned according to the individual with highest fitness value.

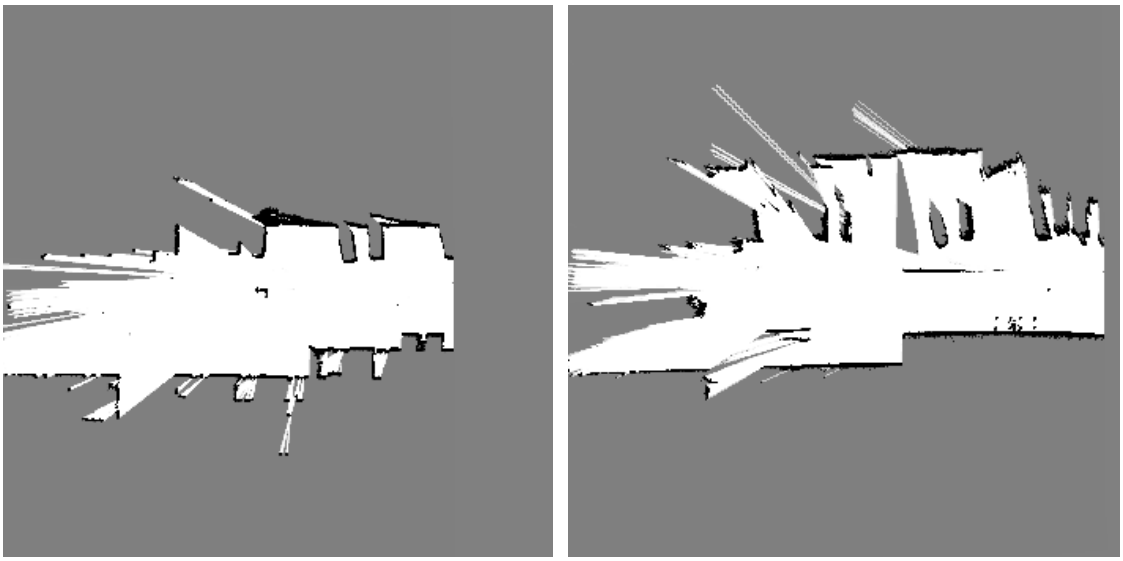


Figure 6.25 The pair of local occupancy grid maps to-be-merged

At the beginning, as illustrated by the Figure 6.26(top-left), a population of individuals was initialized randomly in the error range; the initial maps alignment can be rather erroneous. The Figure 6.26(top-right) and Figure 6.26(bottom-left) display the evolution results before convergence. Finally, after seven rounds of evolution, the correct alignment (the optimal individual) was found, as shown in Figure 6.26(bottom-right); the population was more concentrated around the correct alignment, yet diversity of the population was still maintained.

Then we deliberately changed the optimal individual back to be the initial erroneous maps alignment and also changed each of other individuals by the difference between the initial maps alignment and the optimal individual; we did not provide any information about this change. This is like that the entire population were kidnapped from the correct place to a wrong place, without being informed how they had been displaced or even whether they had been displaced; see the change from Figure 6.26(bottom-right) to Figure 6.27(top-left).

For this population kidnapped, no special procedure was carried out; the genetic evolution process continued in the same way as if no kidnapping had happened; the process is demonstrated in Figure 6.27. As we can see, after several rounds (9 in this case) of evolution, the population recovered from the influence of the kidnapping event and evolved to the correct place, as illustrated in Figure 6.27(bottom-right).

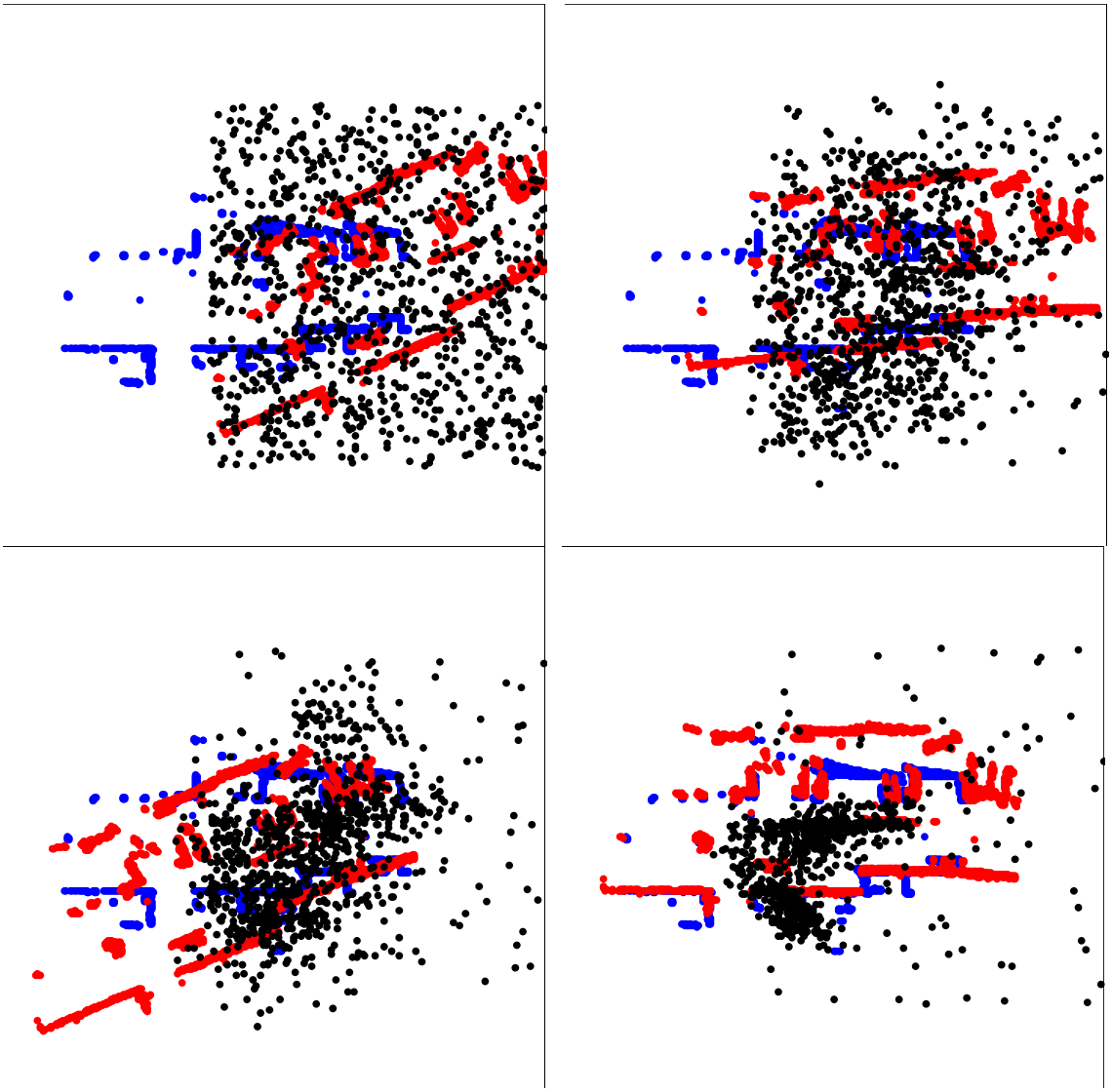


Figure 6.26 Process of genetic evolution

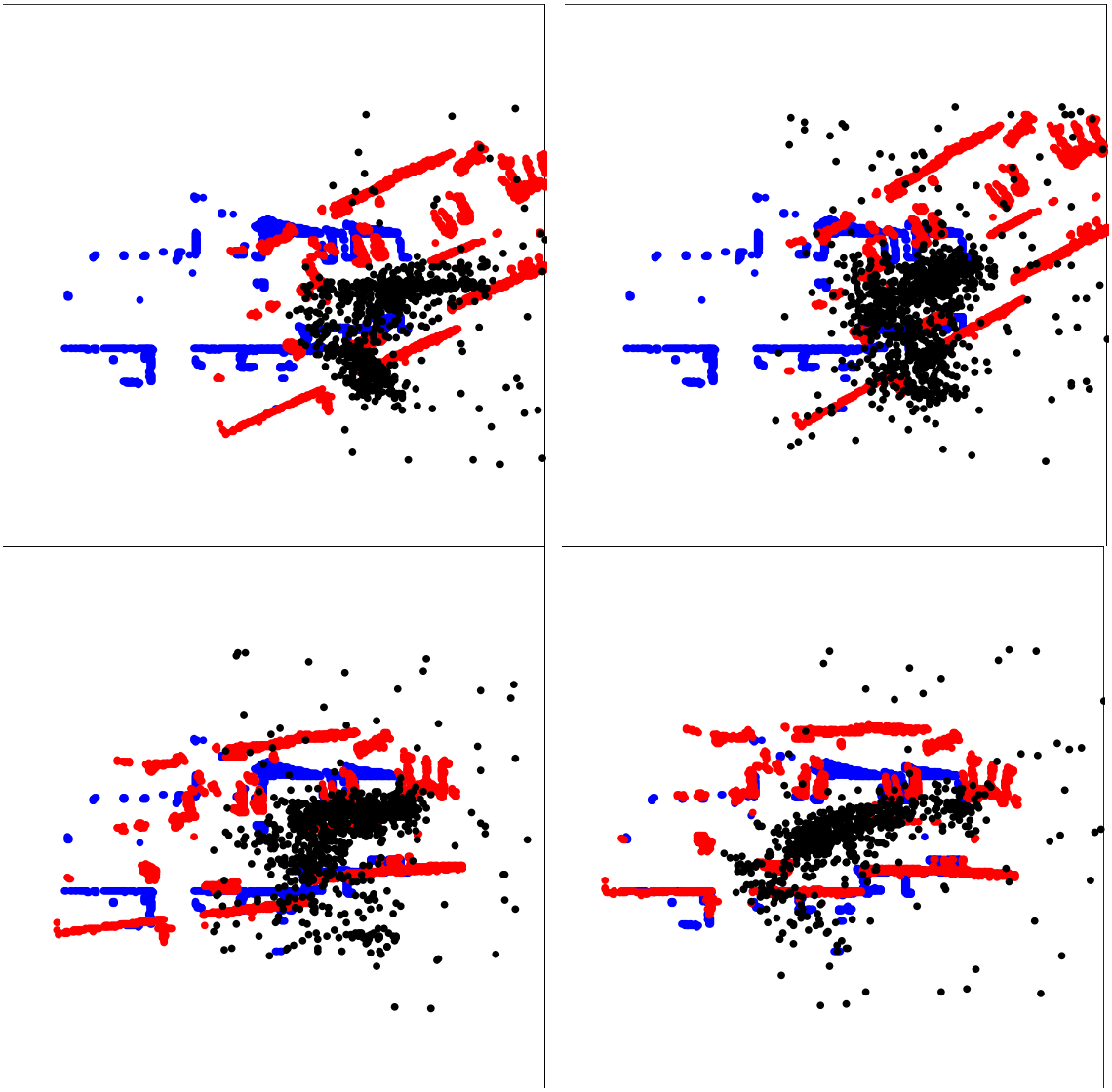


Figure 6.27 Re-convergence (or recovering) from kidnapping

In reality, a kidnapping event hardly happens, yet this experiment demonstrates the potential of the proposed occupancy grid maps merging method to recover the merging result from unexpected misleading factors.

6.4.5 Occupancy Grid Maps Merging: Experiment III

In Section 4.3.2 we propose to use an objective function (4-14) (or (4-13)) for occupancy grid maps merging and extend it into a general formulism in the objective function (4-15) (the objective function (4-12) [Birk & Carpin 2006] is a special case of (4-15) under triplet discretization of the occupancy state space). Nevertheless, we prefer

adhering to (4-14) instead of (4-15), mainly for two reasons. First, we want to avoid any tuning of a heuristic parameter i.e. the c_{lock}. Second, our practices have shown that the objective function (4-15) with a c_{lock} not large enough might incur a wrong solution, whereas the objective function (4-14) or the objective function (4-15) with a large c_{lock} would always have a desirable performance. With a large c_{lock}, the distance-map based similarity term in (4-15) has trivial contribution and can be totally neglected for simplicity. Therefore, we do not need to maintain the complex formulism in (4-15) but just use (4-14), according to the spirit of *Occam's razor*.

In this sub-section, we use some examples of maps inherent inconsistency to show how the objective function (4-15) with a c_{lock} not large enough fails to generate a correct maps alignment.

The first example is generated in simulation and illustrated in Figure 6.28; the top two sub-figures represent two local occupancy grid maps of the same environment. The correct maps alignment is indicated in the bottom-left sub-figure, which also shows the complete appearance of an object. As we can see, there is maps inherent inconsistency between the two local maps; they have a consistent vertical 'T' part, yet the horizontal line segment is pointing left in one map and pointing right in the other—We can imagine that the two local maps are built by two vehicles passing by the object from different sides.

In the occupancy grid maps merging method introduced in Section 4.3, we only changed the objective function into (4-12) with c_{lock} being set to 10 (for expression simplicity, we only refer to (4-12) without always mentioning this value setting of c_{lock}); we kept other implementation conditions unchanged. After several rounds of genetic evolution, an optimal solution was found, as shown in the Figure 6.28 (bottom-right), which is not the correct maps alignment—In order to verify that this wrong optimal solution was not caused by the mal-functioning of the optimization technique, we manually set the maps alignment to be the ground-truth and computed its fitness value. In terms of (4-12), the fitness value of the correct maps alignment was indeed smaller than that of the wrong optimal solution—In contrast, the proposed occupancy grid maps merging method which adopts the objective function (4-14) succeeded in finding the correct maps alignment, as illustrated in Figure 6.28 (bottom-left).

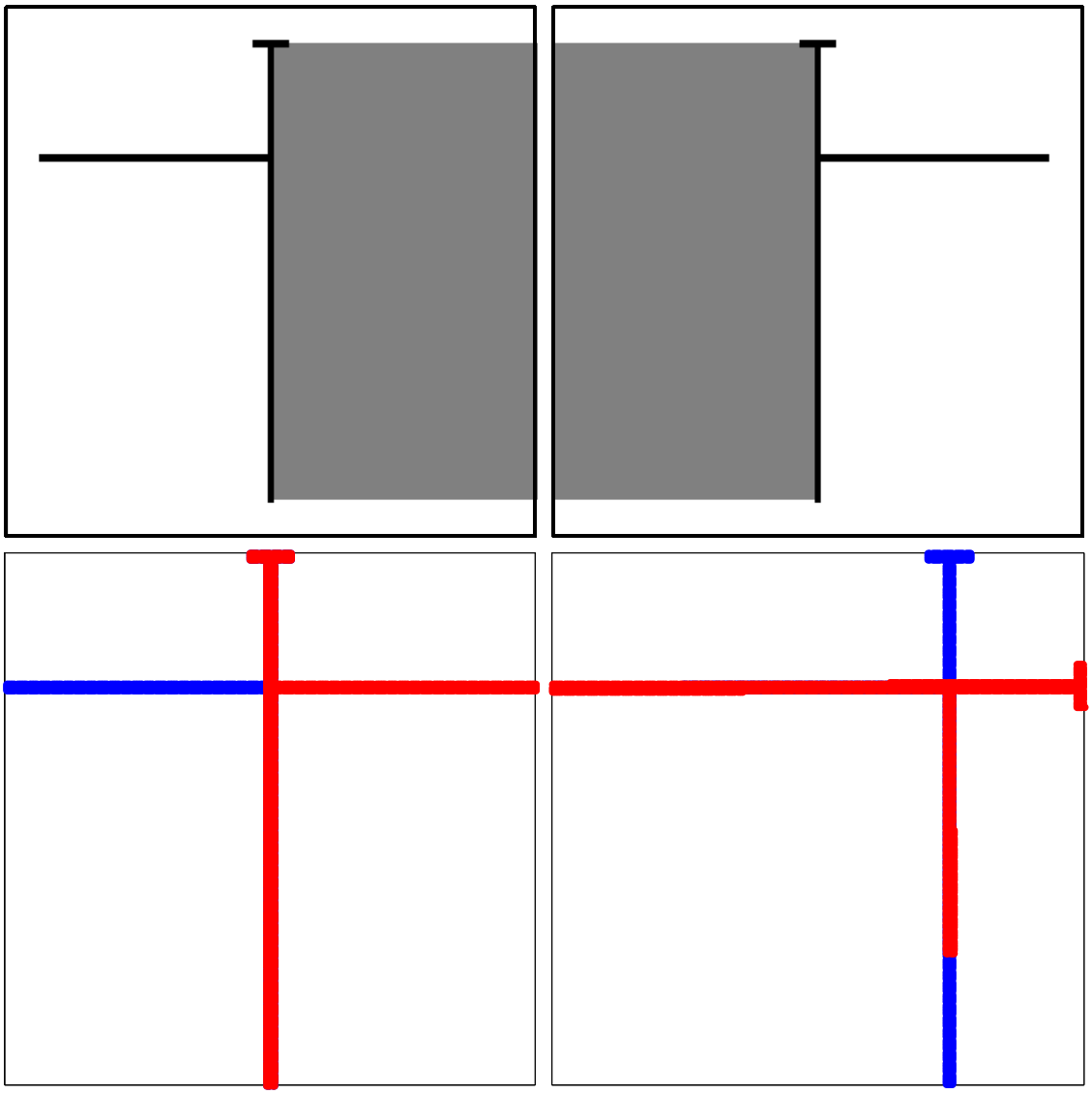


Figure 6.28 Maps inherent inconsistency: synthetic data (top) two local maps; (bottom-left) correct maps alignment; (bottom-right) the wrong alignment

Another example is from real-data and illustrated in Figure 6.29; the top two sub-figures represent two local occupancy grid maps. The objective function (4-12) was again used for the maps merging and a wrong optimal solution was found, as demonstrated in Figure 6.29 (bottom-right). In contrast, the proposed occupancy grid maps merging method which adopts the objective function (4-14) succeeded in finding the correct maps alignment, as demonstrated in Figure 6.29 (bottom-left).

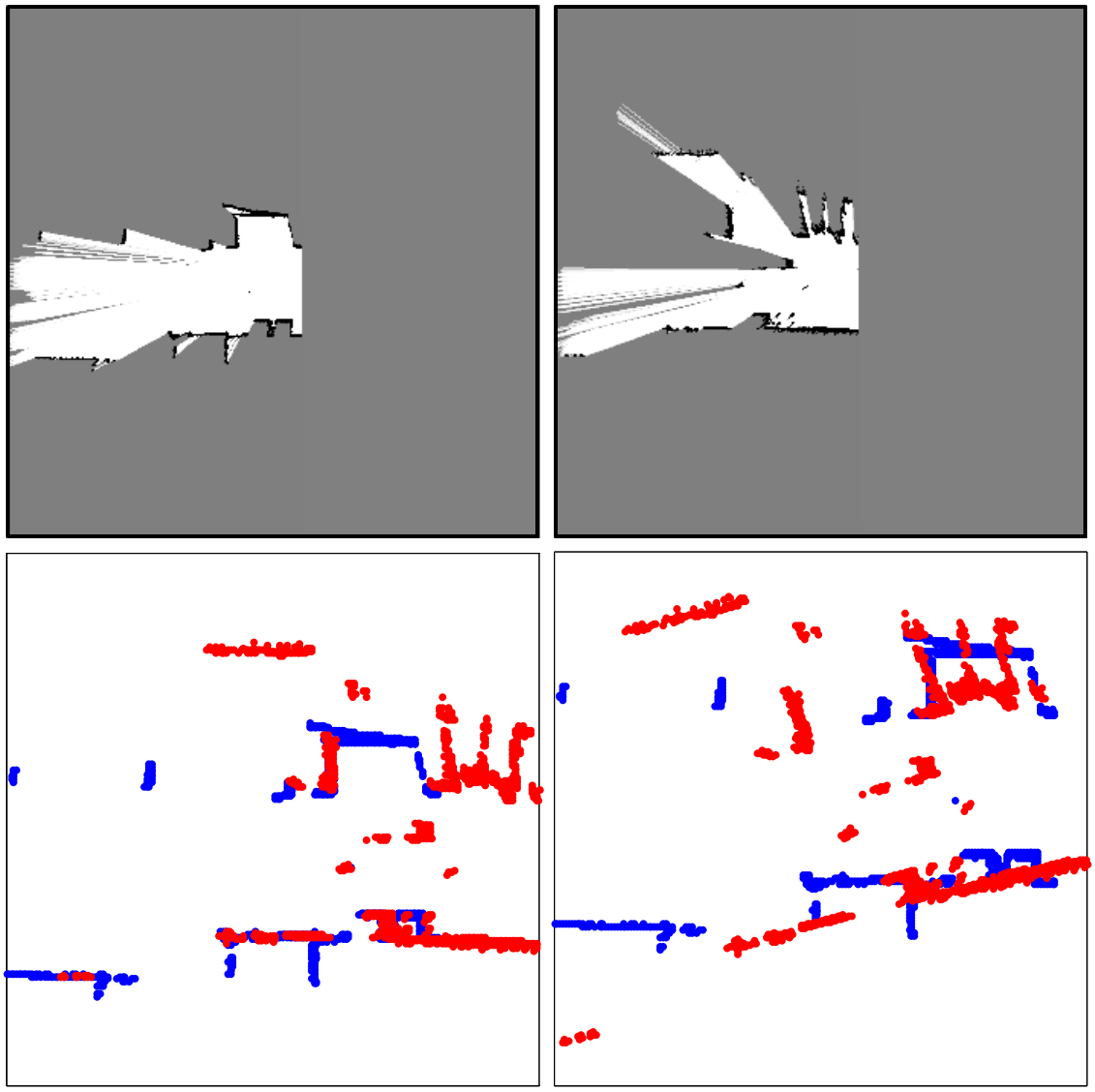


Figure 6.29 Maps inherent inconsistency: real-data (top) two local maps; (bottom-left) correct maps alignment; (bottom-right) the wrong alignment

6.4.6 Cooperative Moving Objects Detection

The effect of cooperative moving objects detection is demonstrated in Figure 6.30. Each of the left two sub-figures shows the local occupancy grid map and detected moving objects of one single vehicle; the detected moving objects are marked by blue boxes. The merged occupancy grid map and moving objects are shown in bottom-right sub-figure. Compared with the bottom-left sub-figure, the bottom-right sub-figure shows a more complete view for the vehicle.

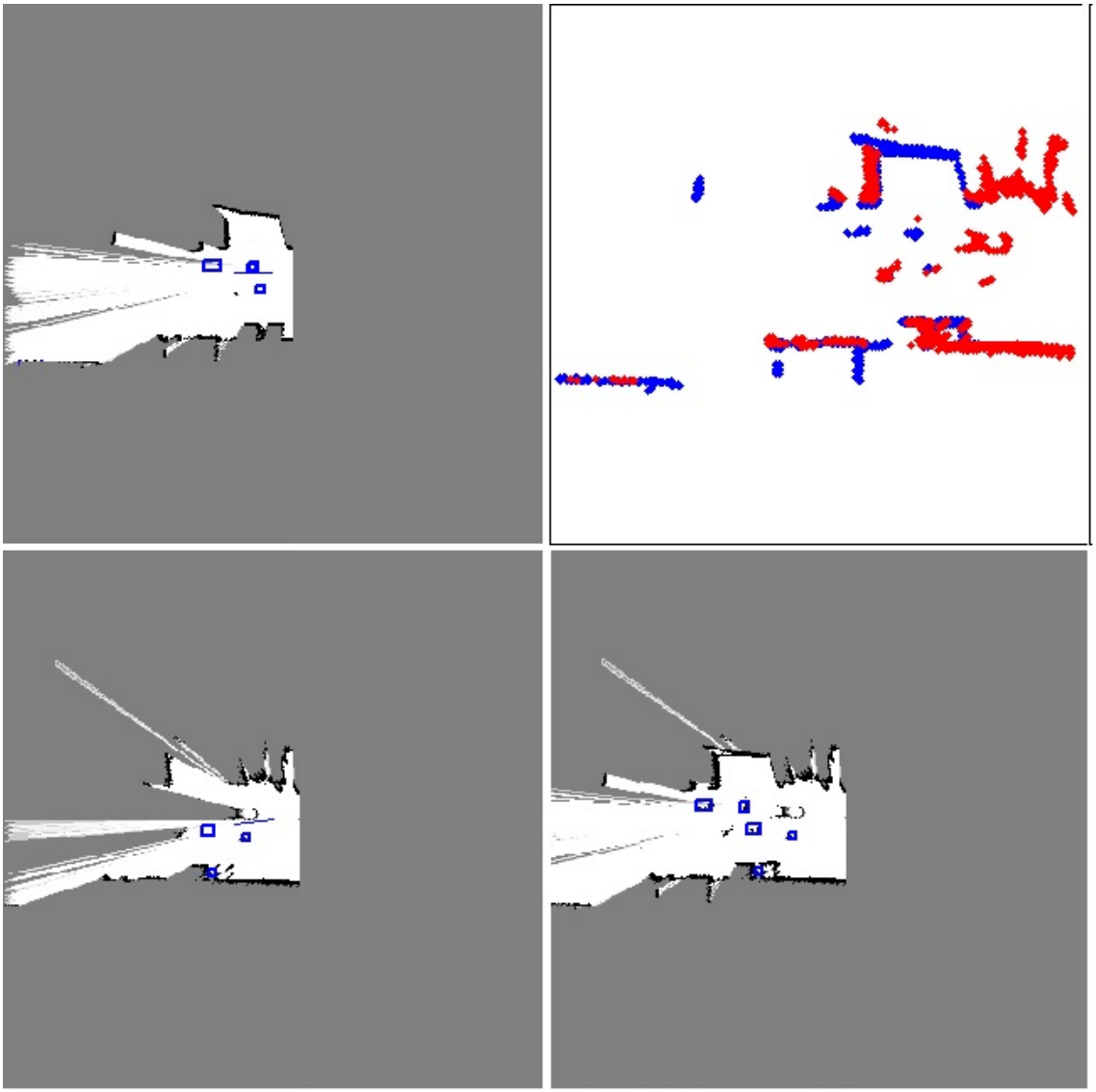


Figure 6.30 Cooperative moving objects detection: (left) local maps and single vehicle moving objects detection; (top-right) local maps merging; (bottom-right) merged moving objects

6.5 Cooperative Augmented Reality

The experimental conditions are the same to those described in Section 6.4, i.e. real data experiments were carried out based on two CyCab vehicle platforms, each of which is equipped with a RTK-GPS, an IBEO laser scanner, and odometer sensors (including steering encoder). The RTK-GPS outputs are intentionally degraded with errors, in order to simulate the situation where only low-cost GPS (comparatively low accuracy)

is available. The time of the two vehicle systems are related to the GPS universal time (a low-cost GPS can also obtain accurate GPS universal time).

In the experiments, one CyCab vehicle was in front and the other was behind, as in the front-following vehicles scenario described in Section 5.2. The sensor coordinates systems of each CyCab vehicle had been pre-calibrated off-line using the method introduced in Section 5.6. During vehicles cooperation, the following vehicle collected the perception of the front vehicle and used the proposed occupancy grid maps merging method to associate the perception of the front vehicle with its own perception. The method introduced in CHAPTER 5 is used to generate an augmented effect of 'seeing' the through front vehicle for the following vehicle. Some performance examples are demonstrated in Figure 2.15 and Figure 6.31 to Figure 6.34.

Each of these examples shows the visual perception taken from the following vehicle in a scenario where the front vehicle occludes partial perception of the following vehicle; the occluded environment is displayed directly on the visual perception of the following vehicle, forming the effect of augmented reality. In spite of distortions at some local places, the effect of augmented reality in these synthetic visual perceptions is lifelike on the whole, as if we can really see through the front vehicle and have a direct and natural visual perception on the occluded environment.

Figure 6.31 Cooperative augmented reality effect: ‘see’ through front vehicle

Figure 6.32 Cooperative augmented reality effect: ‘see’ through front vehicle

Figure 6.33 Cooperative augmented reality effect: ‘see’ through front vehicle

Figure 6.34 Cooperative augmented reality effect: 'see' through front vehicle

6.6 Summary

We have presented the experimental conditions and experimental results concerning cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality. We have presented the results of a simulation based comparative study which demonstrates the advantage of the proposed cooperative localization architecture using split covariance intersection filter, especially for intelligent vehicles with heterogeneous absolute positioning ability. A prominent advantage of the SCIFCL method is that it enables good localization results to be naturally spread within a vehicle network in connection while always keeping a reasonable confidence for the state estimate of each vehicle. We have also presented the results of field tests (real-data) on cooperative localization, which lead to similar conclusions in the simulation based comparative study. We have demonstrated the performance of the proposed occupancy grid maps merging method based on real-data tests. In spite of an intentionally exaggerated initial error range, local occupancy grid

maps built by different vehicles can always be merged correctly using the proposed method; besides, the proposed occupancy grid maps merging method has the potential to recover the merging result from a *kidnapping* situation. We have demonstrated the performance of a proposed method coined as cooperative augmented reality, which realizes a vivid and lifelike effect of 'seeing' through the front vehicle for the following vehicle in a front-following vehicles scenario.

CHAPTER 7 Conclusion

7.1 Dissertation Summary

In the first chapter, we have specified the research theme of this dissertation, i.e. multi-vehicles cooperative perception (or *cooperative perception* for short) applied in the context of intelligent vehicle systems; we have explained our research motivation with two typical traffic scenarios, yet the value of cooperative perception is not limited to them. The general methodology of the presented works in this dissertation is to realize multi-intelligent vehicles cooperative perception, which aims at providing better vehicle perception result compared with single vehicle perception (or non-cooperative perception). Instead of focusing our research works on the absolute performance of cooperative perception, we focus on the general mechanisms which enable the realization of cooperative localization and cooperative mapping (and moving objects detection), considering that localization and mapping are two underlying tasks for an intelligent vehicle system. We also exploit the possibility to realize certain augmented reality effect with the help of basic cooperative perception functionalities; we name this kind of practice as cooperative augmented reality. Naturally, the contributions of the presented works consist in three aspects: cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality.

In CHAPTER 2, we have introduced several sorts of sensors, namely GPS, laser scanner, camera, and motion sensor, which have been commonly used for single intelligent vehicle operation; with these sensors, an intelligent vehicle can possess fairly complete perception ability towards itself and the environment. We have reviewed the Bayesian filter framework that has been commonly used for recursive state estimation; we have also reviewed several recursive estimation methods that are derived from the Bayesian filter framework based on different kinds of approximations. We have discussed in details the fundamental problems and the state-of-the-art methods concerned in cooperative localization, and cooperative local mapping and moving objects detection. Based on these discussions, we propose a general architecture of cooperative localization using split covariance intersection filter, an indirect vehicle-to-vehicle relative pose estimation method, and a new method for occupancy grid maps merging to handle the fundamental problems in cooperative localization, and

cooperative local mapping and moving objects detection. We propose a brand new idea of cooperative augmented reality which utilizes cooperative perception results to realize a special augmented effect.

In CHAPTER 3, we have provided a solution of multi-vehicles **c**ooperative **l**ocalization. We have reviewed the concept of estimate consistency and the split covariance intersection filter; we have presented several forms of this filter together with their derivations and an original proof for the fusion consistency of this filter. We have specified the compounding notation for coordinate transformation and explained some properties of this compounding notation. We have introduced several basic functionalities as the condition for realizing cooperative localization; these functionalities are abstracted from field practice based on their feasibility in reality. We have described a general architecture of cooperative localization using split covariance intersection filter; as the architecture is decentralized, we have described from the perspective of an intelligent vehicle how it can evolve its state estimate using its motion measurements, how it can update its state estimate using its own absolute positioning measurements, and how it can update its state estimate with the data shared by neighbouring vehicles. We have presented the indirect vehicle-to-vehicle relative pose estimation strategy.

In CHAPTER 4, we have provided a solution of **c**ooperative **lo**cal **m**apping and **m**oving **o**bjects **d**etection for laser scanner based intelligent vehicles. We have reviewed the method of occupancy grid based single vehicle local SLAM, including how to use laser scanner based range measurements to incrementally update the occupancy grid map estimate according to the inverse measurement model and how to estimate current vehicle local state (pose) with last estimate of vehicle local state and occupancy grid map. We have explained the different roles of vehicle local state and vehicle global state; we have described how vehicle local state estimate in SLAM can be used to assist vehicle global state estimation. We have presented the framework for occupancy grid maps merging by generalizing its essential part into an optimization problem; we have proposed a new objective function that measures the consistency degree of maps alignment based on occupancy likelihood. We have adopted the spirit of genetic algorithm and designed a set of concrete procedures to search the optimal maps alignment. We have introduced the scheme of multi-vehicles cooperative moving objects detection based on occupancy grid maps merging; for a complete implementation, we have reviewed two basic moving objects detection methods, namely the consistency-based detection and the moving object map based detection.

In CHAPTER 5, we have extended the spirit of augmented reality to cooperative perception, forming the concept of ***c****ooperative* ***a****ugmented* ***r****eality* in the context of intelligent vehicle systems. We have specified the front-following vehicles scenario to which the proposed idea of cooperative augmented reality is applied. We have reviewed the pinhole camera model and described how to establish spatial relationship between two views (easily extendable to multi-views case) according to perspective geometry. We have described several coordinates systems i.e. the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system that are concerned in an intelligent vehicle; we have introduced a technique of utilizing a 2D laser scanner to assist a mono-camera in estimating the visual perception depth approximately. We have presented how to map the visual perception of a vehicle onto that of another vehicle, abiding by the multi-views perspective geometry described. We have also introduced a new extrinsic calibration method for a camera and a 2D laser scanner, which can reveal all the spatial relationships among the camera coordinates system, the laser scanner coordinates system, the ground coordinates system, and the vehicle coordinates system, based only on the popular chessboard calibration practice with few extra measurements.

In CHAPTER 6, we have presented the experimental conditions and experimental results concerning cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality. We have presented the results of a simulation based comparative study which demonstrates the advantage of the proposed cooperative localization architecture using split covariance intersection filter (SCIFCL), especially for intelligent vehicles with heterogeneous absolute positioning ability. A prominent advantage of the SCIFCL method is that it enables good localization results to be naturally spread within a vehicle network in connection while always keeping a reasonable confidence for the state estimate of each vehicle. We have also presented the results of field tests (real-data) on cooperative localization, which lead to similar conclusions in the simulation based comparative study. We have demonstrated the performance of the proposed occupancy grid maps merging method based on real-data tests. In spite of an intentionally exaggerated initial error range, local occupancy grid maps built by different vehicles can always be merged correctly using the proposed method; besides, the proposed occupancy grid maps merging method has the potential to recover the merging result from a *kidnapping* situation. We have demonstrated the performance of a proposed method coined as cooperative augmented reality, which realizes a vivid and lifelike effect of ‘seeing’ through the front vehicle for the following vehicle in a front-following vehicles scenario.

To conclude in few words, we have proposed some cooperative perception methods which enable an intelligent vehicle to take advantage of the data from other vehicles, for the purpose of refining its global state estimate, the purpose of complementing its inference about the environment, and the purpose of generating vivid and lifelike visualization effect.

7.2 Perspectives

This dissertation has presented some works concerning cooperative perception and their applications in the context of outdoor intelligent vehicle systems. More works can further be done concerning this topic. Here, we discuss several valuable research directions for future extensions or improvements.

7.2.1 Thorough Fusion of Environment State Estimates

In current works, an intelligent vehicle always maintains an independent environment state estimate (mapping) and associate it with the independent environment state estimates of other vehicles; however, the merged environment state estimate which incorporates those of other vehicles is only used for current time and will not be used during following operation. In other words, the environment state estimates of other vehicles are not thoroughly fused into that of the ego vehicle. This practice is to guarantee the independence among the environment state estimates of different vehicles and further guarantee the consistency of fusion results.

This is similar to the practice in [Karam et al. 2006b] which handles the estimation of vehicles states instead of environment states. In this dissertation, we have introduced a general architecture of cooperative localization using split covariance intersection filter, which is flexible to fuse various sources of data while always keeping the consistency of vehicle state estimate. Unfortunately, we can not adopt the split covariance intersection filter for environment states fusion in current works in a similar way. Why?

First, a basic requirement for applying the split covariance intersection filter is the availability of posterior uncertainty. However, for each vehicle, we do not maintain posterior uncertainty for its environment state estimate; the local map estimated is only a map with maximum likelihood in certain recursive sense.

Second, the environment state usually has huge dimensions. Direct application of the split covariance intersection filter on high dimensional environment state is computationally forbidding.

Therefore, future research works are needed to find a methodology which enables a vehicle to thoroughly fuse the environment state estimates of other vehicles into its own.

7.2.2 General Architecture of C-SLAMMOT (Cooperative Simultaneous Localization and Mapping with Moving Objects Tracking)

The techniques of moving objects detection and tracking are not exploited deeply in current works, partially because the local maps merging method (*Cooperation*) presented and the moving objects detection method adopted are rather independent of each other. In fact, even the relation between the SLAM (for one single vehicle) and the cooperation is also mutually independent: each vehicle always performs local SLAM independently; the local maps built by different vehicles are merged only instantly but not fused thoroughly (as explained in Section 7.2.1).

As pointed out by Wang [Wang 2004], SLAM and MOT (moving objects tracking, including detection) are mutually beneficial and had better be handled together, forming the concept of SLAMMOT; this spirit was put forward in practice by Vu [Vu 2009]. Similarly, this spirit can be extended to multi-vehicles cooperation: SLAM and MOT, together with Cooperation, are also mutually beneficial.

SLAM and MOT are beneficial to Cooperation: First, with better local maps, Cooperation would become more robust and accurate. Second, moving objects can be treated as special feature which might facilitate Cooperation.

On the other hand, Cooperation is also beneficial to SLAM and MOT. With Cooperation, more complete map can be generated for an intelligent vehicle, which can make the SLAM more robust and accurate. With cooperation, occlusions which are undesirable factors for MOT can be largely reduced.

Therefore, a direction is to put the processes of SLAM, MOT, and Cooperation into a general architecture, forming an integrated process of C-SLAMMOT. For **distributed** realization, this is not easy, yet worthy future research works.

APPENDIX

I. Coordinate Transformation: Compounding Operation

We follow the compounding notation in [Wang 2004]:

$$\begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \end{bmatrix} \oplus \begin{bmatrix} x_2 \\ y_2 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} x_2 \cos\theta_1 - y_2 \sin\theta_1 + x_1 \\ x_2 \sin\theta_1 + y_2 \cos\theta_1 + y_1 \\ \theta_2 + \theta_1 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \end{bmatrix} \oplus \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_2 \cos\theta_1 - y_2 \sin\theta_1 + x_1 \\ x_2 \sin\theta_1 + y_2 \cos\theta_1 + y_1 \end{bmatrix}$$

$$inv\left(\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}\right) = \begin{bmatrix} -x\cos\theta - y\sin\theta \\ x\sin\theta - y\cos\theta \\ -\theta \end{bmatrix}$$

We can easily verify the associativity of the compounding notation i.e.

$$\mathbf{T}_1 \oplus (\mathbf{T}_2 \oplus \mathbf{T}_3) = (\mathbf{T}_1 \oplus \mathbf{T}_2) \oplus \mathbf{T}_3$$

We can also verify that $\mathbf{0}$ i.e. $[0, 0, 0]^T$ is the only identity element for the compounding operation, which satisfies

$$\mathbf{I} \oplus \mathbf{T} = \mathbf{T} \Leftrightarrow \mathbf{I} = \mathbf{0}$$

We can also verify that each non-identity element $\mathbf{T}$ has the only inverse element $inv(\mathbf{T})$:

$$\mathbf{A} \oplus \mathbf{T} = \mathbf{T} \oplus \mathbf{A} = \mathbf{0} \Leftrightarrow \mathbf{A} = inv(\mathbf{T})$$

The compounding notation is commonly used for coordinate transformation. For example, given a coordinates system parametrized as **T** in a global reference; then for an arbitrary point (or pose) **p** in **T**, its global representation is $\mathbf{T} \oplus \mathbf{p}$.

BIBLIOGRAPHY

[Abuhadrous *et al.* 2003] I. Abuhadrous, F. Nashashibi, C. Laurgeau, M. Chinchole, "Multi-sensor data fusion for land vehicle localization using ^{RT}MAPS", *IEEE Intelligent Vehicle Symposium*, 2003, pp.339-344

[Ammoun *et al.* 2007] S. Ammoun, F. Nashashibi, C. Laurgeau. "An analysis of the lane changing manoeuvre on roads: the contribution of inter-vehicle cooperation via communication", *IEEE Intelligent Vehicles Symposium*, 2007, pp.1095-1100

[Bailey & Whyte 2006] T. Bailey, H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II", *IEEE Robotics & Automation Magazine*, 13(3), 2006, pp.108-117

[Bertozzi & Broggi 1998] M. Bertozzi, A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection", *IEEE Trans on Image Processing*, 7(1), 1998, pp.62-81

[Besl & McKay 1992] P.J. Besl, N.D. McKay, "A method for registration of 3-D shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 1992, pp.239-256

[Biber & Strasser 2003] P. Biber, W. Strasser, "The normal distributions transform: A new approach to laser scan matching", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp.2743-2748

[Birk & Carpin 2006] A. Birk, S. Carpin, "Merging occupancy grids from multiple robots", *Proceedings of the IEEE*, 94(7), 2006, pp.1384-1397

[Bruin *et al.* 2004] D. Bruin, J. Kroon, R. Klaveren, M. Nelisse, "Design and test of a cooperative adaptive cruise control system", *IEEE Intelligent Vehicles Symposium*, 2004, pp.392-396

[Carlone *et al.* 2011] L. Carlone, M.K. Ng, J. Du, B. Bona, M. Indri, "Simultaneous localization and mapping using rao-blackwellized particle filters in multi robot systems", *Journal of Intelligent and Robotic Systems*, 63(2), 2011, pp.283-307

[Chan & Bougler 2005] C.Y. Chan, B. Bougler, "Evaluation of cooperative roadside and vehicle-based data collection for assessing intersection conflicts", *IEEE Intelligent vehicles Symposium*, 2005, pp.165-170

[Charrette & Nashashibi 2009] R. de Charrette, F. Nashashibi, "Real time visual traffic lights recognition based on spot light detection and adaptive traffic lights templates", *IEEE Intelligent Vehicles Symposium*, 2009, pp.358-363

[Chen & Medioni 1992] Y. Chen, G.G. Medioni, "Object modeling by registration of multiple range images", *Image and Vision Computing*, 10(3), 1992, pp.145-155

[Cox 1990] I.J. Cox, "Blanche: Position estimation for an autonomous robot vehicle", *In: Autonomous Robot Vehicles*, *I.J. Cox and G.T.Wilfong (eds), Springer-Verlag*, 1990

[Cox 1991] I. Cox, "Blanche – An experiment in guidance and navigation of an autonomous robot vehicle", *IEEE Transactions on Robotics and Automation*, 7(2), 1991, pp.193-204

[Dedeoglu & Sukhatme 2000] G. Dedeoglu, G. Sukhatme, "Landmark-based matching algorithm for cooperative mapping by autonomous robots", *International Symposium on Distributed Autonomous Robotics Systems*, 2000, pp.251-260

[Doucet *et al.* 2000] A. Doucet, S. Godsill, C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering", *Statistics and Computing*, 10(3), 2000, pp.197-208

[Doucet *et al.* 2001] A. Doucet, N. De Freitas, N. Gordon (eds.), "Sequential Monte Carlo methods in practice", *New York: Springer-Verlag*, 2001

[Elfes 1989] A. Elfes, "Occupancy grids: a probabilistic framework for robot perception and navigation", *PhD Thesis, Carnegie Mellon University*, 1989

[Enzweiler & Gavrila 2009] M. Enzweiler, D.M. Gavrila, "Monocular pedestrian detection: survey and experiments", *IEEE Transactions on Pattern Analysis and Machine Learning*, 31(12), 2009, pp.2179-2195

[Farahmand & Mili 2009] A.S. Farahmand, L. Mili, "Cooperative decentralized intersection collision avoidance using extended kalman filter", *IEEE Intelligent Vehicles Symposium*, 2009, pp.977-982

[Farrell & Barth 1998] J. Farrell, M. Barth, "The global positioning system and inertial navigation", *New York: McGraw-Hill*, 1998

[Faugeras 1993] O. Faugeras, "Three-dimensional computer vision: a geometric approach", *Cambridge, MA:MIT Press*, 1993

[Fox *et al.* 2000] D. Fox, W. Burgard, H. Kruppa, S. Thrun, "A probabilistic approach to collaborative multi-robot localization", Autonomous Robots, 8(3), 2000, pp.325-344

[Gate 2009] G. Gate, "Reliable perception of highly changing environments: implementations for car-to-pedestrian collision avoidance systems", *PhD Thesis, Ecole Nationale Superieure des Mines de Paris*, 2009

[Gate *et al.* 2009] G. Gate, A. Breheret, F. Nashashibi, "Centralized fusion for fast people detection in dense environment", *IEEE International Conference on Robotics and Automation*, 2009, pp.76-81

[Gil *et al.* 2010] A. Gil, O. Reinoso, M. Ballesta, M. Julia, "Multi-robot visual SLAM using a Rao-Blackwellized particle filter", *Robotics and Autonomous Systems* 58, 2010, pp.68-80

[Golub & Loan 1996] G. Golub, C. van Loan, "Matrix Computations", *Johns Hopkins University Press, Baltimore, Maryland, 3rd edition*, 1996

[Grewal & Andrews 2000] M.S. Grewal, A.P. Andrews, "Kalman filtering: theory and practice", *2nd ed., Wiley, New York*, 2000

[Grewal *et al.* 2001] M.S. Grewal, L.R. Weill, A.P. Andrews, "Global Positioning Systems, Inertial navigation and integration", *John Wiley & Sons*, 2001

[Grossmann & Poli 2001] A. Grossmann, R. Poli, "Robust mobile robot localization from sparse and noisy proximetry readings using Hough transform and probability grids", *Robotics and Autonomous Systems*, 37, 2001, pp.1-18

[Guivant *et al.* 2000] J. Guivant, E. Nebot, H. Durrant-Whyte, "Simultaneous localization and map building using natural features in outdoor environments", *IAS-6 Intelligent Autonomous Systems*, 2000, pp.581-586

[Gutmann & Knolige 1999] J.S.Gutmann, K. Konolige, "Incremental mapping of large cyclic environments", Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA), 1999, pp.318-325

[Gutmann & Schlegel 1996] J.S. Gutmann, C. Schlegel, "AMOS: Comparison of scan matching approaches for self-localization in indoor environments", *Proceedings of the first Euromicro Workshop on Advanced Mobile Robot*, 1996, pp.61-67

[Howard 2006] A. Howard, "Multi-robot simultaneous localization and mapping using particle filters", International Journal of Robotics Research, 25(12), 2006, pp.1243-1256

[Howard *et al.* 2002] A. Howard, M.J. Mataric, G.S. Sukhatme, "Localization for mobile robot teams using maximum likelihood estimation", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, pp.434-439

[Howard *et al.* 2003] A. Howard, M.J. Mataric, G.S. Sukhatme, "Putting the 'I' in 'team': an ego-centric approach to cooperative localization", *IEEE International Conference on Robotics and Automation*, vol.1, 2003, pp.868-874

[Howard *et al.* 2006] A. Howard, L.E. Parker, G.S. Sukhatme, "The SDR experience: Experiments with a large-scale heterogeneous mobile robot team", *Experimental Robotics IX, STAR 21, Springer-Verlag Berlin Heidelberg*, 2006, pp.121-130

[Julier & Uhlmann 1997] S.J. Julier, J.K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations", *Proceedings of American Control Conference*, 1997, pp.2369-2373

[Julier & Uhlmann 2001] S.J. Julier, J.K. Uhlmann, "General decentralized data fusion with covariance intersection (CI)", *Chapter 12, D. Hall, J. Llinas (Eds.), Handbook of Data Fusion, CRC Press, Boca Raton FL USA*, 2001

[Kalman 1960] R.E. Kalman, "A new approach to linear filtering and prediction problem", *ASME Transactions, Series D: Journal of Basic Engineering*, 82, 1960, pp.35-45

[Kao 1991] W.W. Kao, "Integration of GPS and dead-reckoning navigation systems", *Vehicle Navigation and Information Systems Conference*, vol.2, 1991, pp.635-643

[Karam *et al.* 2006a] N. Karam, F. Chausse, R. Aufrere, R. Chapuis, "Cooperative multi-vehicle localization", *IEEE Intelligent Vehicles Symposium*, 2006, pp.564-570

[Karam *et al.* 2006b] N. Karam, F. Chausse, R. Aufrere, R. Chapuis, "Localization of a group of communicating vehicles by state exchange", *IEEE International Conference on Intelligent Robots and Systems*, 2006, pp.519-52

[Khaled *et al.* 2009] Y. Khaled, M. Tsukada, J. Santa, J. Choi, T. Ernst, "A usage oriented analysis of vehicular networks: from technologies to applications", *Journal of Communications*, 4(5), 2009, pp.357-368

[Khaled *et al.* 2010] Y. Khaled, M. Tsukada, J. Santa, T. Ernst, "The role of communication and network technologies in vehicular applications", *Book Chapter 15, Advances in Vehicular Ad-Hoc Networks: Developments and Challenges, IGI Global*, 2010

[Koenders & Vreeswijk 2008] E. Koenders, J. Vreeswijk, "Cooperative infrastructure", *IEEE Intelligent Vehicles Symposium*, 2008, pp.721-726

[Labayrade *et al.* 2005] R. Labayrade, C. Royere, D. Gruyer, D. Aubert, "Cooperative fusion for multi-obstacle detection with use of stereovision and laser scanner", *Autonomous Robot*, 19(2), 2005, pp.117-140

[Laneurit *et al.* 2005] J. Laneurit, R. Chapuis, F. Chausse, "Accurate vehicle positioning on a numerical map", *International Journal of Control, Automation, and Systems*, 3(1), 2005, pp.15-31

[Leung *et al.* 2009] K.Y.K. Leung, T.D. Barfoot, H.H.T. Liu, "Decentralized localization for dynamic and sparse robot networks", IEEE International Conference on Robotics and Automation, 2009, pp.3135-3141

[Li & Nashashibi 2011a] H. Li, F. Nashashibi,"Multi-vehicle cooperative perception and augmented reality for driver assistance: a possibility to 'see' through front vehicle", IEEE International Conference on Intelligent Transportation Systems, 2011, pp.242-247

[Li & Nashashibi 2011b] H. Li, F. Nashashibi, "Robust real-time lane detection based on lane mark segment features and general *a priori* knowledge", *IEEE International Conference on Robotics and Biomimetics*, 2011, pp.812-817

[Li & Nashashibi 2012a] H. Li, F. Nashashibi, "Cooperative multi-vehicle localization using split covariance intersection filter", *IEEE Intelligent Vehicles Symposium*, 2012, pp.211-216

[Li & Nashashibi 2012b] H. Li, F. Nashashibi, "Multi-vehicle cooperative localization using indirect vehicle-to-vehicle relative pose estimation", *IEEE International Conference on Vehicular Electronics and Safety*, 2012

[Li & Nashashibi 2012c] H. Li, F. Nashashibi, "A new method for occupancy grid maps merging: Application to multi-vehicle cooperative local mapping and moving

object detection in outdoor environment", ***to appear in*** *International Conference on Control, Automation, Robotics and Vision*, 2012

[Li & Wang 2006] L. Li, F.Y. Wang. "Cooperative driving at blind crossing using intervehicle communication", *IEEE Transactions on Vehicular Technology*, 55(6), 2006, pp.1712-1724

[Li *et al.* 2005] L. Li, F.Y. Wang, H. Kim, "Cooperative driving and lane changing at blind crossings", *IEEE Intelligent Vehicles Symposium*, 2005, pp.435-440

[Li *et al.* 2010] H. Li, F. Nashashibi, G. Toulminet, "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data", *IEEE International Conference on Intelligent Transportation Systems*, 2010, pp.1657-1662

[Lingemann *et al.* 2005] K. Lingemann, A. Nuchter, J. Hertzberg, H. Surmann, "High-speed laser localization for mobile robots", *Robotics and Autonomous Systems* 51, 2005, pp.275-296

[Lu & Milios 1997a] F. Lu, E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans", *Journal of Intelligent and Robotic Systems*, 18, 1997, pp.249-275

[Lu & Milios 1997b] F. Lu, E. Milios, "Globally consistent range scan alignment for environment mapping", *Autonomous Robots*, 4, 1997, pp.333-349

[Madhavan *et al.* 2004] R. Madhavan, K. Fregene, L.E. Parker, "Distributed cooperative outdoor multirobot localization and mapping", *Autonomous Robots*, 17(1), 2004, pp.23-39

[Man *et al.* 1999] K.F. Man, K.S. Tang, S. Kwong, "Genetic algorithms", *New York: Springer-Verlag*, 1999

[Minguez *et al.* 2006] J. Minguez, L. Montesano, F. Lamiraux, "Metric-based Iterative Closest Point scan matching for sensor displacement estimation", *IEEE Transactions on Robotics*, 22(5), 2006, pp.1047-1054

[Montemerlo *et al.* 2002] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem", *Proceedings of the AAAI National Conference on Artificial Intelligence*, 2002, pp.593-598

[More 1977] J.J. More, "The Levenberg-Marquardt algorithm: implementation and theory", *In G.A. Watson editor, Numerical Analysis, Lecture Notes in Mathematics, Springer-Verlag* 630, 1977, pp.105-116

[Moutarde *et al.* 2007] F. Moutarde, A. Bargeton, A. Herbin, L. Chanussot, "Robust on-vehicle real-time visual detection of American and European speed limit signs, with a modular traffic signs recognition system", *IEEE Intelligent Vehicles Symposium*, 2007, pp.1122-1126

[Murphy 2002] K.P. Murphy, "Dynamic Bayesian Networks: representation, inference and learning", *PhD Thesis, University of California, Berkeley*, 2002

[Nerurkar *et al.* 2009] E.D. Nerurkar, S.I. Roumeliotis, A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization", *IEEE International Conference on Robotics and Automation*, 2009, pp.1402-1409

[Piao & McDonald 2008] J. Piao, M. McDonald. "Potential applications of road-vehicle communication for improving safety and mobility", *Road Transport Information and Control & ITS United Kingdom Members' Conference*, 2008, pp.1-6

[Pomerleau 1989] D.A. Pomerleau, "ALVINN: An autonomous land vehicle in a neural network", *Technical Report CMU-CS-89-107*, Carnegie Mellon Univ., 1989

[Rajamani 2005] R. Rajamani, "Vehicle dynamics and control", *New York: Springer-Verlag*, 2005

[Raza & Ioannou 1996] H. Raza, P. Ioannou. "Vehicle following control design for automated highway systems", *IEEE Control Systems Magzine*, 16(6), 1996, pp.43-60

[Redmill *et al.* 2001] K.A. Redmill, T. Kitajima, U. Ozgumer, "DGPS/INS integrated positioning for control of automated Vehicle", *Proc IEEE Intelligent Transportation Systems*, 2001, pp.172-178

[Rezaei & Sengupta 2007] S. Rezaei, R. Sengupta, "Kalman filter-based integration of DGPS and vehicle sensors for localization", *IEEE Trans on Control Systems Technology*, 15(6), 2007, pp.1080-1088

[Roumeliotis2002] S.I. Roumeliotis, G.A. Bekey, "Distributed multirobot localization", *IEEE Transactions on Robotics and Automation*, 18(5), 2002, pp.781-795

[Royer 2006] E. Royer, “Cartographie 3D et localisation par vision monoculaire pour la navigation autonome d’un robot mobile”, *PhD Thesis, Université Blaise Pascal-Clermont II*, 2006

[Rusconi *et al.* 2007] G. Rusconi, M.C. Brugnoli, P. Dosso, K. Kretzschmar, et al, “I-WAY, intelligent co-operative system for road safety”, *IEEE Intelligent Vehicles Symposium*, 2007, pp.1056-1061

[Rusinkiewicz & Levoy 2001] S. Rusinkiewicz, M. Levoy, “Efficient variants of the ICP algorithm”, *3rd International Conference on 3D Digital Imaging and Modeling*, 2001, pp.145-152

[Saeedi *et al.* 2011] S. Saeedi, L. Paull, M. Trentini, H. Li, “Multiple Robot Simultaneous Localization and Mapping”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp.853-858

[Skog & Handel 2009] I. Skog, P. Handel, “In-car positioning and navigation technologies-a survey”, *IEEE Transactions on Intelligent Transportation Systems*, 10(1), 2009, pp.4-21

[Sun *et al.* 2006] Z. Sun, G. Bebis, R. Miller, “On-road vehicle detection: a review”, *IEEE Transactions on Pattern Analysis and Machine Learning*, 28(5), 2006, pp.694-711

[Tan & Huang 2006] H.S. Tan, J. Huang, “DGPS-based vehicle-to-vehicle cooperative collision warning: engineering feasibility viewpoints”, *IEEE Transactions on Intelligent Transportation Systems*, 7(4), 2006, pp.415-428

[Thorpe *et al.* 1988] C. Thorpe, M.H. Hebert, T. Kanade, S.A. Shafer, “Vision and navigation for the Carnegie-Mellon Navlab”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3), 1988, pp.362-373

[Thrun 2003] S. Thrun, “Learning occupancy grids with forward sensor models”, *Autonomous Robots* 15, 2003, pp.111-127

[Thrun *et al.* 2005] S. Thrun, W. Burgard, D. Fox, “Probabilistic robotics”, *MIT Press*, 2005

[Topal *et al.* 2010] S. Topal, I. Erkmen, A.M. Erkmen, “A novel map merging methodology for multi-robot systems”, *Proceedings of the World Congress on Engineering and Computer Science*, 2010, pp.383-387

[Tsugawa *et al.* 2000] S. Tsugawa, S. Kato, T. Matsui, H. Naganawa, H. Fujii, "An architecture for cooperative driving of automated vehicles", *IEEE International Conference on Intelligent Transportation Systems*, 2000, pp.422-427

[Tsugawa *et al.* 2001] S. Tsugawa, S. Kato, K. Tokuda, T. Matsui, H. Fujii, "A cooperative driving system with automated vehicles and inter-vehicle communications in Demo 2000", *Proc IEEE Intelligent Transportation Systems*, 2001, pp.918-923

[Tsugawa *et al.* 2007] S. Tsugawa, S. Kato, N. Hashimoto, N. Minobe, M. Kawai, "Elderly driver assistance systems with cooperation between vehicles: the concept and experiments", *IEEE Intelligent Vehicles Symposium*, 2007, pp.668-672

[Vahidi & Eskandarian 2003] A. Vahidi, A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control", *IEEE Transactions on Intelligent Transportation Systems*, 4(3), 2003, pp.143-153

[Vu 2009] T.D. Vu, "Vehicle perception: Localization, mapping with detection, classification and tracking of moving objects", *PhD Thesis, Institut National Polytechnique de Grenoble*, 2009

[Wang 2004] C.C. Wang, "Simultaneous Localization, Mapping and Moving Object Tracking", *PhD Thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh PA*, 2004

[Wang *et al.* 2003] C.C. Wang, C. Thorpe, S. Thrun, "Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas", *IEEE International Conference on Robotics and Automation*, vol.1, 2003, pp.842-849

[Whyte & Bailey 2006] H. Durrant-Whyte, T. Bailey, "Simultaneous localization and mapping: part I", *IEEE Robotics & Automation Magazine*, 13(2), 2006, pp.99-110

[Xu *et al.* 2000] J. Xu, G. Chen, M. Xie, "Vision-guided automatic parking for smart car", *IEEE Intelligent Vehicles Symposium*, 2000, pp.725-730

[Zhang & Pless 2004] Q. Zhang, R. Pless, "Extrinsic calibration of a camera and laser range finder (improves camera calibration)", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, pp.2301-2306

[Zhang 2000] Z. Zhang, "A flexible new technique for camera calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 2000, pp.1330-1334

Perception Coopérative: Application au Contexte des Systèmes de Véhicules Intelligents à l'Extérieur

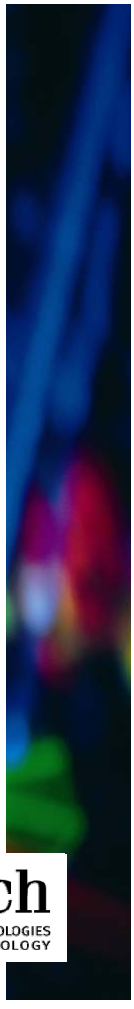
RESUME : Le thème de recherche de cette thèse est la perception coopérative multi-véhicules appliquée au contexte des systèmes de véhicules intelligents. L'objectif général des travaux présentés dans cette thèse est de réaliser la perception coopérative de plusieurs véhicules (dite « perception coopérative »), visant ainsi à fournir des résultats de perception améliorés par rapport à la perception d'un seul véhicule (ou « perception non-coopérative »). Au lieu de concentrer nos recherches sur la performance absolue de la perception coopérative, nous nous concentrons sur les mécanismes généraux qui permettent la réalisation de la localisation coopérative et de la cartographie de l'environnement routier (y compris la détection des objets), considérant que la localisation et la cartographie sont les deux tâches les plus fondamentales pour un système de véhicule intelligent. Nous avons également exploité la possibilité d'explorer les techniques de la réalité augmentée, combinées aux fonctionnalités de perception coopérative. Nous baptisons alors cette approche « réalité augmentée coopérative ». Par conséquent, nous pouvons d'ores et déjà annoncer trois contributions des travaux présentés: la localisation coopérative, la cartographie locale coopérative, et la réalité augmentée coopérative.

Mots clés : perception coopérative, fusion de données, véhicule intelligent, localisation, cartographie, détection

Cooperative Perception: Application in the Context of Outdoor Intelligent Vehicle Systems

ABSTRACT : The research theme of this dissertation is the multiple-vehicles cooperative perception (or cooperative perception) applied in the context of intelligent vehicle systems. The general methodology of the presented works in this dissertation is to realize multiple-intelligent vehicles cooperative perception, which aims at providing better vehicle perception result compared with single vehicle perception (or non-cooperative perception). Instead of focusing our research works on the absolute performance of cooperative perception, we focus on the general mechanisms which enable the realization of cooperative localization and cooperative mapping (and moving objects detection), considering that localization and mapping are two underlying tasks for an intelligent vehicle system. We also exploit the possibility to realize certain augmented reality effect with the help of basic cooperative perception functionalities; we name this kind of practice as cooperative augmented reality. Naturally, the contributions of the presented works consist in three aspects: cooperative localization, cooperative local mapping and moving objects detection, and cooperative augmented reality.

Keywords : cooperative perception, data fusion, intelligent vehicle, localization, mapping, detection



MINES ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY